# Common Criteria for Information Technology Security Evaluation

CCEB-96/012

## Part 2 : Security functional requirements

Version 1.0

96/01/31

# Foreword

Following extensive international cooperation to align the source criteria from Canada (CTCPEC), Europe (ITSEC) and the United States of America (TCSEC and Federal Criteria), version 1.0 of the *Common Criteria for Information Technology Security Evaluation* is issued for the purpose of trial evaluations and for review by the international security community. The practical experience acquired through trial evaluations and all the comments received will be used to further develop the criteria.

A template for reporting observations on version 1.0 of the CC is included at the end of the annexes of this document. Any observation reports should be communicated to one or more of the following points of contact at the sponsoring organisations:


**National Institute of Standards and Technology**
Computer Security Division
NIST North Building, Room 426
Gaithersburg, Maryland 20899
U.S.A.
Tel: (+1)(301)975-2934, Fax:(+1)(301)926-2733
E-mail:csd@nist.gov
http://csrc.ncsl.nist.gov

**National Security Agency**
Attn: V2, Common Criteria Technical Advisor
Fort George G. Meade, Maryland 21122
U.S.A.
Tel: (+1)(410)859-4458, Fax:(+1)(410)684-7512
E-mail: common_criteria@radium.ncsc.mil

**Communications Security Establishment**
Criteria Coordinator
R2B IT Security Standards and Initiatives
P.O. Box 9703, Terminal
Ottawa, Canada K1G 3Z4
Tel:(+1)(613)991-7409, Fax:(+1)(613)991-7411
E-mail:criteria@cse.dnd.ca
ftp:ftp.cse.dnd.ca
http://www.cse.dnd.ca

**UK IT Security and Certification Scheme**
Senior Executive
P.O. Box 152
Cheltenham GL52 5UF
United Kingdom
Tel: (+44) 1242 235739, Fax:(+44)1242 235233
E-mail: ccv1.0@itsec.gov.uk
ftp: ftp.itsec.gov.uk
http://www.itsec.gov.uk

**Bundesamt für Sicherheit in der Informationstechnik**
Abteilung V
Postfach 20 03 63
D-53133 Bonn
Germany
Tel: (+49)228 9582 300, Fax:(+49)228 9582 427
E-mail:cc@bsi.de

**Service Central de la Sécurité des Systèmes d'Information**
Bureau Normalisation, Critères Communs
18 rue du docteur Zamenhof
92131 Issy les Moulineaux
France
Tel: (+33)(1)41463784, Fax:(+33)(1)41463701
E-mail:ssi28@calvacom.fr

**Netherlands National Communications Security Agency**
P.O. Box 20061
NL 2500 EB The Hague
The Netherlands
Tel: (+31) 70 3485637, Fax:(+31).70.3486503
E-mail: criteria@nlncsa.minbuza.nl


This document is paginated from i to x and from 1 to 178

# Table of contents

## List of figures

# List of tables

# Chapter 1

# Introduction

## 1.1 Scope

1 Security functional components, as defined in this Part 2, are the basis for the TOE IT security functional requirements expressed in a Protection Profile (PP) or a Security Target (ST). These requirements describe the desired security behaviour expected of a Target of Evaluation (TOE) and are intended to meet the security objectives as stated in a PP or an ST. These requirements describe security properties that users can detect by direct interaction with the TOE (i.e., inputs, outputs) or by the TOE's response to stimulus.

2 Security functional components express security requirements intended to counter threats in the assumed operating environment of the TOE or comply with organisational security policies.

3 The audience for Part 2 includes consumers, developers, and evaluators of secure IT systems and products. Part 1 chapter 1 provides additional information on the target audience of the Common Criteria (CC), and on the use of the CC by the target audience. These groups may use Part 2 as follows:

- Consumers may use Part 2 when selecting components to express requirements to satisfy the security objectives in order to counter identified threats in their operational environment expressed in a PP or ST. Part 1 chapter 2 provides more detailed information on the relationship between security objectives to threats and security requirements.

- Developers, who respond to actual or perceived consumer security requirements in constructing a TOE, may find a standardised method to understand those requirements in this part. They can also use the contents of this part as a basis for further defining the TOE security functions and mechanisms that comply with those requirements.

- Evaluators should use the functional requirements defined in this part of the CC in verifying that the TOE functional requirements expressed in the PP or ST satisfy the IT security objectives and that all dependencies are accounted for and shown to be satisfied. Evaluators also should use this part to assist in determining whether a given TOE satisfies stated requirements.

### 1.1.1 Extended functional requirements

4 The CC and the associated security functional requirements described herein are not meant to be a definitive answer to all the problems of IT security. Rather, the CC offers a set of well understood security functional requirements which can be used to create trusted products or systems reflecting the needs of the market. These

security functional requirements are presented as the current state of the art in requirements specification and evaluation.

5 This part does not presume to include all possible security functional requirements but rather contains those which are known and agreed to be of value by the CC sponsoring organisations at the time of release.

6 If it can be shown by an ST author that a new security functional requirement provides additional protection against a specific threat or offers useful functions not currently provided by the CC, then the evaluation authority of any of the sponsoring organisations shall determine the applicability, strength, and utility of the new functional requirement. This evaluation authority will indicate to the ST author whether, in the context of the CC, such a security functional requirement is appropriate and how the evaluation of the security functional requirement will be carried out.

### 1.1.2 Status of Cryptographic Support Requirements

7 Cryptography is an important and powerful mechanism for the implementation of IT security functions, particularly in networked and distributed system architectures. Owing to the late development of criteria related to cryptographic functions, and therefore their relative immaturity in relation to the rest of the document, this version of the CC does not contain requirements specific to cryptography or cryptographic mechanisms. However, it is acknowledged that this is an important technology area that must be addressed in the next version of the CC.

8 The draft material, created from Federal Information Processing Standard (FIPS) 140-1 and the Canadian Trusted Computer Product Evaluation Criteria (CTCPEC) and intended for inclusion in the CC, is available for public review and use as a "work in progress". This material can be found in ***Cryptographic Support Requirements,*** a draft submission for the Common Criteria version 1.0.

## 1.2 Organisation of Part 2

9 Chapter 1 is the introductory material for Part 2.

10 Chapter 2 is the catalogue of CC functional components.

11 Chapter 3 is the catalogue of CC functional packages. No packages have been defined in this version of the CC.

12 Annex A provides additional information of interest to potential users of the functional components. It is a repository for informative supporting material for the users of this part, which may help them to apply relevant operations and select appropriate audit or documentation information.

13 Annex B is a repository for informative supporting material for the users of this part which presents a threat-based approach for selecting functional families and components during development of PPs and STs.

14 Annex C provides the Common Criteria observation report guidance, example observations and an example printed form.

15 These annexes, A through C, are not included in this document but are contained in a separate, companion document.

16 Those who author PPs or STs should refer to Part 1 for relevant structures, rules, and guidance:

- Part 1, Annex A defines the terms used in the CC.
- Part 1, Annex B defines the structure for PPs.
- Part 1, Annex C defines the structure for STs.

## 1.3 Functional requirements paradigm

17 This section describes the paradigm used in the security functional requirements of Part 2. Figures 1.1 and 1.2 depict some of the key concepts of the paradigm. This section provides descriptive text for those figures and for other key concepts not depicted. Key concepts discussed are highlighted in bold/italics. This section is not

intended to replace or supersede any of the terms found in the CC glossary in Part 1, Annex A.

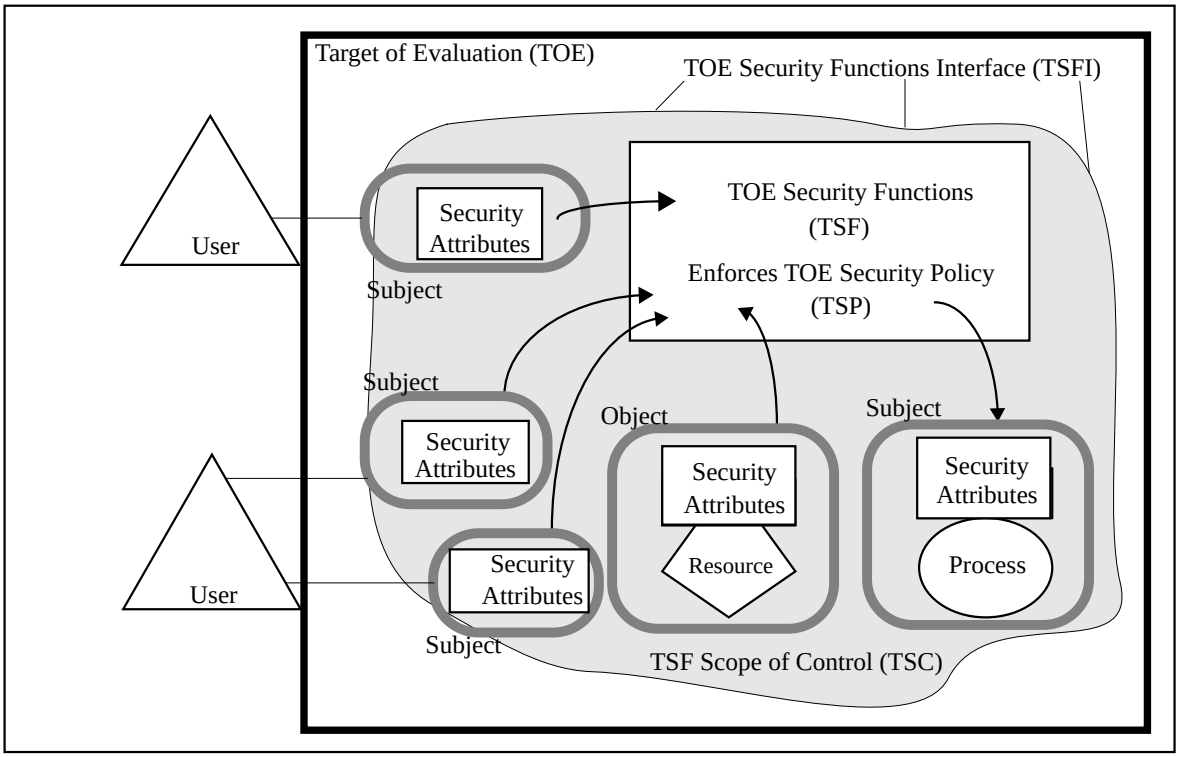



**Figure 1.1 - Security functional requirements paradigm (Monolithic TOE)**

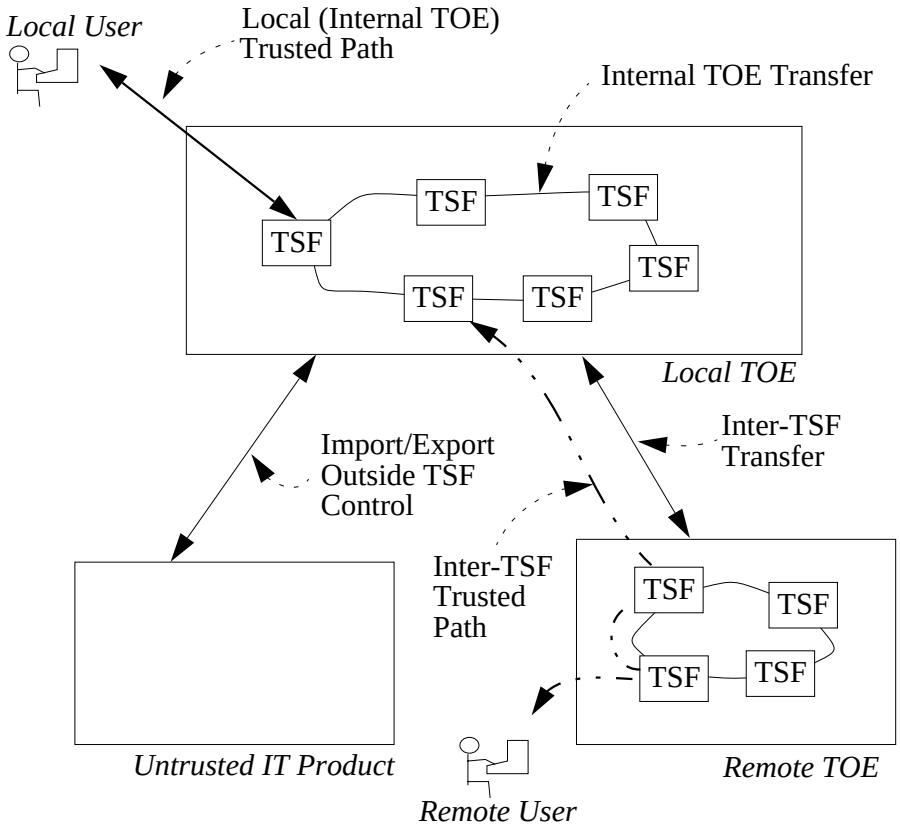



**Figure 1.2 - Diagram of security function requirements paradigm (Distributed TOE)**

18 Part 2 is a catalogue of security functional requirements which can be specified for a ***Target of Evaluation (TOE)***. A TOE is an IT product or system containing resources such as electronic storage media (e.g., disks), peripheral devices (e.g., printers), and computing capacity (e.g., CPU time) that can be used for processing and storing information and is the subject of an evaluation.

19 TOE evaluation is concerned primarily with ensuring that a defined ***TOE Security Policy (TSP)*** is enforced over the TOE resources. The TSP defines the rules by which the TOE governs access to its resources, and thus all information and services controlled by the TOE.

20 The TSP is, in turn, made up of multiple ***Security Function Policies (SFPs)***. Each SFP has a scope of control, which defines the subjects, objects, and operations controlled under the SFP. The SFP is implemented by one or more ***Security Functions (SFs)***, whose mechanisms enforce the policy and provide necessary capabilities.

21 There are two types of data protection SFPs: ***access control SFPs*** and ***information flow SFPs***. The mechanisms that implement access control SFPs base their policy decisions on ***attributes*** of the subjects, objects, and operations within the scope of control. These attributes are used in the rules that govern operations that subjects may perform on objects.

22 The mechanisms that implement information flow SFPs base their policy decisions on the security attributes assigned to subjects and objects within the scope of control and the set of rules that govern the transfer of information between subjects and objects.

23 All the portions of a TOE which must be relied on for the correct enforcement of the TSP (i.e., this collection of multiple SFPs) is referred to as the ***TOE Security Functions (TSF)***. The TSF consists of all hardware, software, and firmware of a TOE which either directly enforces or contributes to the enforcement of the TSP.

24 A ***reference monitor*** is an abstract machine that enforces the access control policies of a TOE. A ***reference validation mechanism*** is an implementation of the reference monitor concept that possesses the following properties: tamperproof, always invoked, and small enough to be subjected to thorough analysis and testing. The ***TSF*** consists of a reference validation mechanism (potentially) and other functions necessary for the operation of the TOE.

25 The TOE may be a monolithic product containing hardware, firmware, and software, or it may consist internally of multiple physically-separated parts. Each of these parts of the TOE provides a particular service for the TOE, and is connected to the other parts of the TOE through an ***internal communication channel***. This channel can be as small as a processor bus, or may encompass a network internal to the TOE.

26 When the TOE consists of multiple parts, each part of the TOE may have its own part of the TSF. When the TOE is viewed as a whole, the separate parts of the TSF abstractly form the composite TSF, which enforces the TSP. In order to do this, the parts of the TOE exchange user and TSF data over internal communication channels. This interaction is called ***internal TOE transfer***.

27 TOE interfaces may be localised to the particular TOE, or they may allow interaction with other TOEs over ***external communication channels***. These external interactions with other TOEs may take two forms:

a) The TSPs of the remote and local TOEs have been administratively coordinated and evaluated. Exchanges of information in this situation are called ***inter-TSF transfers***, as they are between the TSFs of distinct TOEs.

b) The remote TOE may not have been evaluated, therefore its TSP is unknown. Exchanges of information in this situation are called ***transfers outside TSF control***, as there is no TSF (or its policy characteristics are unknown) on the remote TOE.

28 The set of interactions which can occur with or within a TOE and are subject to the rules of the TSP is called the ***TSF Scope of Control (TSC)***. The TSC encompasses a defined set of interactions based on subjects, objects, and operations within the TOE, but it need not encompass all resources of a TOE.

29 The set of interfaces, whether interactive (man-machine interface) or programmatic (application programming interface), through which resources are accessed or information is obtained from the TSF, is referred to as the ***TSF Interface (TSFI)***. The TSFI defines the boundaries of the TOE functions which provide for the enforcement of the TSP.

30 Users are outside of the TOE, and therefore outside the TSC. However, in order to request that services be performed by the TOE, users interact with the TOE through the TSFI. There are two types of users of interest to the Part 2 security functional requirements, ***human users*** and ***machine users***. Human users are further differentiated as ***local human users***, meaning they interact directly with the TOE via TOE devices (e.g., workstations), or ***remote human users***, meaning they interact indirectly with the TOE through another TOE.

31 A period of interaction between users and the TSF is referred to as a user ***session***. Establishment of user sessions can be controlled based on a variety of considerations including, for example: user authentication, time of day, method of accessing the TOE, and number of allowed concurrent sessions per user.

32 Part 2 uses the term ***authorised*** to signify a user who possesses the rights and/or privileges necessary to perform an operation. The term ***authorised user***, therefore, indicates that it is allowable for a user to perform an operation as defined by the TSP.

33 The term ***authorised administrator*** is used to indicate a human user who is trusted to perform security critical operations within the TOE, such as setting TOE configuration parameters that may affect the enforcement of the TSP, and therefore possesses the specific rights necessary to perform those operations.

When the term “the authorised administrator” is used in the security functional components, it is referring specifically to an administrator authorised with respect to the SFP related to the functions in the component. When the more general phrase “authorised administrators” is used, it refers to administrators who are authorised for SFPs other than the one related to the functions in question.

34 To express requirements that call for the separation of administrator duties, the relevant Part 2 security functional components (from family FPT_TSA) explicitly state that administrative ***roles*** are required. A role is a pre-defined set of allowed authorisations that may be granted to a user. A TOE may support the definition of any number of roles. For example, roles related to the secure operation of a TOE may include “Audit Administrator” and “User Accounts Administrator”. Roles may also be defined specifically for the application environment in which the TOE will be used. For example, in a TOE used in a hospital, a “Doctor” role might be established for users who are authorised to prescribe medication. However, any user

operating in a “Nurse” role may only be authorised to administer any such medication. Roles required by the components in Part 2 are security related.

35 TOEs contain resources which may be used for the processing and storing of information. The primary goal of the TSF is the complete and correct enforcement of the TSP over the resources and information that the TOE controls.

36 TOE resources can be structured and utilised in many different ways. However, Part 2 makes a specific distinction which allows for the specification of desired security properties.

37 All entities which can be created from resources can be characterised in one of two ways. The entities may be active, meaning that they are the cause of actions which occur internal to the TOE and cause operations to be performed on information. Alternatively, the entities may be passive, meaning that they are either the container from which information originates or to which information is stored.

38 Active entities are referred to as ***subjects***. Several types of subjects may exist within a TOE:

a) those acting on behalf of an authorised user and which are subject to all the rules of the TSP (e.g., UNIX processes);

b) those acting as a specific functional process which may in turn act on behalf of multiple users (e.g., client/server architecture);

c) those acting on behalf of authorised administrators; or

d) those acting as part of the TOE itself (e.g., trusted processes).

39 Part 2 addresses the enforcement of the TSP over the type of subjects listed above.

40 Passive entities (i.e., information containers) are referred to in the Part 2 security functional requirements as ***objects***. Objects are the targets of operations that may be performed by subjects. In the case where a subject (an active entity) is the target of an operation (e.g., interprocess communication), a subject may also be acted on as an object.

41 Both subjects and objects possess certain ***attributes*** which contain information that allows the TOE to behave correctly. Some attributes, such as file names, may be intended to be informational (i.e., to increase the user-friendliness of the TOE) while others, such as access control information, may exist specifically for the enforcement of the TSP. These latter attributes are referred to as security attributes. However, no matter what the intended purpose of the attribute information, it may be necessary to have controls on attributes as dictated by the TSP.

42 Data in a TOE is categorised as either user data or TSF data. Figure 1.3 depicts this relationship. ***User Data*** is information stored in TOE resources that can be operated upon by users in accordance with the TSP and upon which the TSF places no special meaning. For example, the contents of an electronic mail message is user data. ***TSF***

***Data*** is information used by the TSF in making TSP decisions. TSF Data may be influenced by users if allowed by the TSP. Security attributes, authentication data and access control list entries are examples of TSF data.

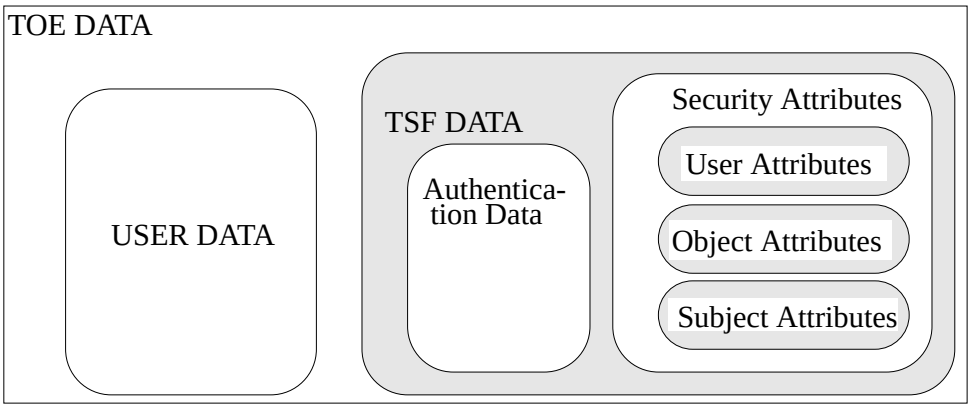



**Figure 1.3 - Relationship Between User Data and TSF Data**

43 Two specific types of TSF data addressed by Part 2 can be, but are not necessarily, the same. These are ***authentication data*** and ***secrets***.

44 Authentication data is used to verify the claimed identity of a user requesting services from a TOE. The most common form of authentication data is the password, which depends on being kept secret in order to be an effective security mechanism. However, not all forms of authentication data need to be kept secret. Biometric authentication devices (e.g., fingerprint readers, retinal scanners) do not rely on the fact that the data is kept secret, but rather that the data is something that only one user possesses and that it cannot be forged.

45 The term secrets, as used in CC functional requirements, while applicable to authentication data, is intended to also be applicable to other types of data that must be kept secret in order to enforce a specific SFP. For example, a trusted channel mechanism that relies on cryptography to preserve the confidentiality of information being transmitted via the channel can only be as strong as the method used to keep the cryptographic keys secret from unauthorised disclosure.

46 Therefore, some, but not all, authentication data needs to be kept secret and some, but not all, secrets are used as authentication data. Figure 1.4 shows this relationship between secrets and authentication data.

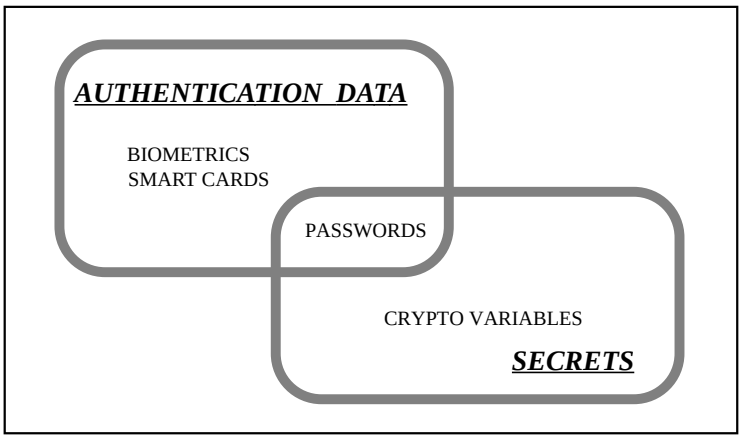



**Figure 1.4 - Relationship between "authentication data" and "secrets".**

# Chapter 2

# Security functional components

## 2.1 Overview

47 This section defines the content and presentation of the functional requirements of the CC and provides guidance on the organisation of the requirements for new components to be included in a security target and to be evaluated. The functional requirements are expressed in classes, families, and components.

### 2.1.1 Class structure

48 Figure 2.1 illustrates the functional class structure in diagrammatic form.

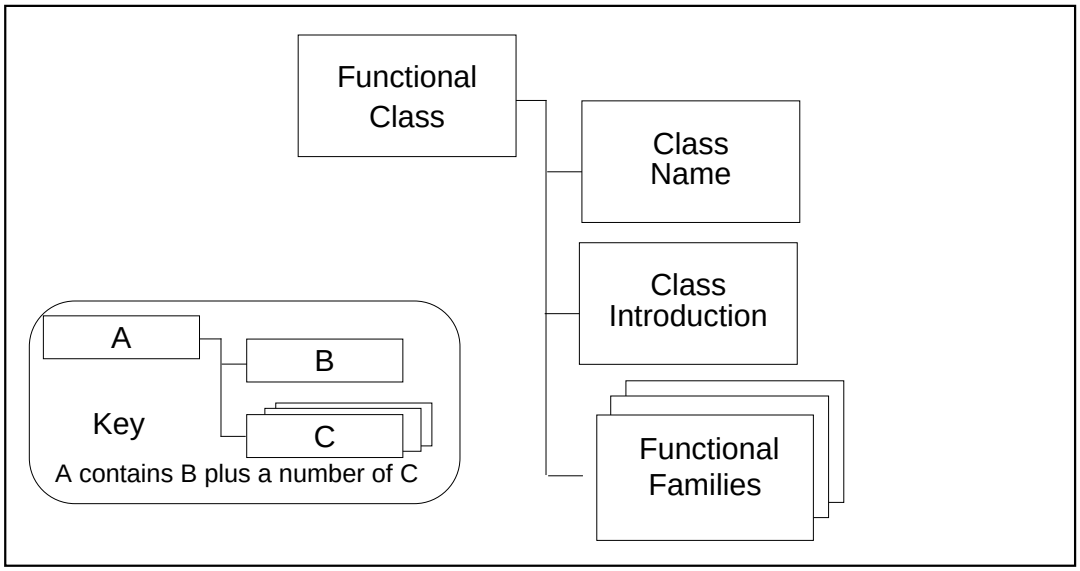



**Figure 2.1 - Functional class structure.**

#### 2.1.1.1 Class name

49 The class name section provides information necessary to identify and categorise a functional class. Every functional class has a unique name. The categorical information consists of a short name of three characters. The short name of the class is used in the specification of the short names of the families of that class.

#### 2.1.1.2 Class introduction

50 The class introduction expresses the common intent or approach of those families to satisfy security objectives. The definition of functional classes does not reflect any formal taxonomy in the specification of the requirements.

51 The class introduction provides a figure describing the families in this class and the hierarchy of the components in each family.

### 2.1.2 Family structure

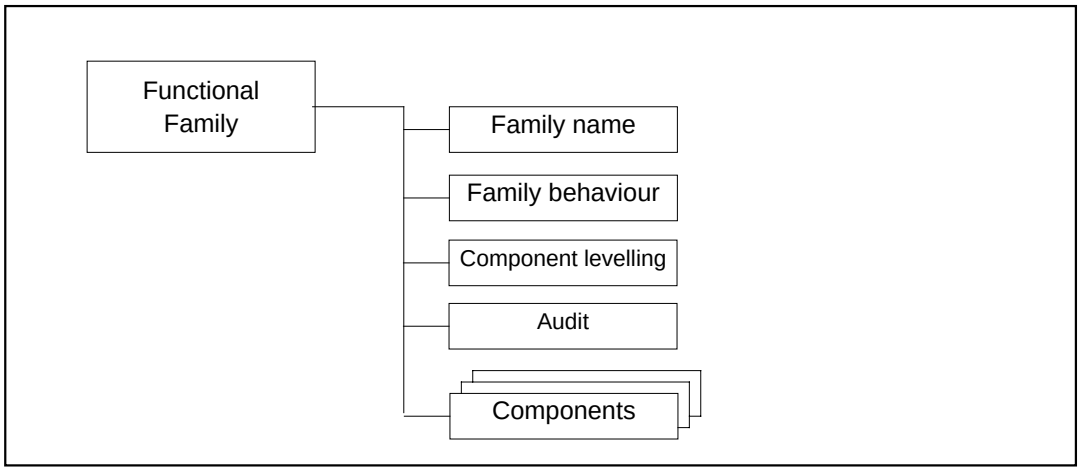



**Figure 2.2 - Functional family structure**

52 Figure 2.2 illustrates the functional family structure in diagrammatic form.

#### 2.1.2.1 Family name

53 The family name section provides categorical and descriptive information necessary to identify and categorise a functional family. Every functional family has a unique name. The categorical information consists of a short name of seven characters, with the first three identical to the short name of the class followed by an underscore and the short name of the family as follows XXX_YYY. The unique short form of the family name provides the principal reference name for the components.

#### 2.1.2.2 Family behaviour

54 The family behaviour is the narrative description of the functional family stating its security objective, and a general description of the functional requirements. These are described in greater detail below:

a) The *security objectives* of the component family is a clear and concise statement of the security problem for which a TOE incorporating a component belonging to the family contributes to the solution.

b) The description of the *functional requirements* summarises all the requirements that are included in the component(s). The description is aimed at authors of PPs, STs, and functional packages who wish to assess whether the family is relevant to their specific requirements.

#### 2.1.2.3 Component levelling

55 Functional families contain one or more components, any one of which can be selected for inclusion in PPs, STs, and functional packages. The goal of this section is to provide information to users in selecting an appropriate functional component once the family has been identified as being a necessary or useful part of their security requirements.

56 This section of the functional family description describes the components available, their rationale, and the relationships between components. The exact details of the components are contained within each component. For a family with only one component, this part of the functional family contains a statement to the effect that the family currently contains only one component.

57 The relationships between components within a functional family may or may not be hierarchical. A component is hierarchical to another if it offers more functionality, for example, the TOE offers additional functions or offers the existing functions to additional users. Thus it may not be a superset of the previous component or it may not be more secure, as it may introduce additional potential vulnerabilities.

#### 2.1.2.4 Audit

58 The *audit* requirements contain information for the PP/ST authors to select auditable events if requirements from the class FAU, Security Audit are included in the PP/ST. These requirements include security relevant events in terms of the various levels of detail supported by the components of the FAU_GEN Security Audit Data Generation family. For example, an audit note might include actions that are in terms of: Minimal - successful use of the security mechanism; Basic - any use of the security mechanism as well as relevant information regarding the security attributes involved; Detailed - any configuration changes made to the mechanism, including the actual configuration values before and after the change.

### 2.1.3 Component structure

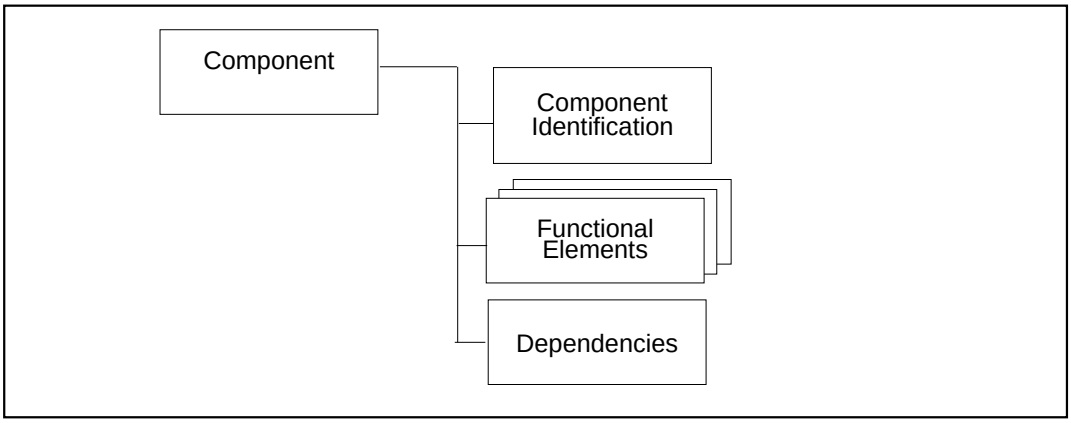



**Figure 2.3 - Functional component structure**

59 Figure 2.3 illustrates the functional component structure.

#### 2.1.3.1 Component identification

60 The component identification section provides descriptive information necessary to identify, categorise, register, and cross-reference a component. The following is provided as part of every functional component:

61 A *unique name*. The name reflects the purpose of the component.

62 A *short name*. A unique short form of the functional component name. This short name serves as the principal reference name for the categorisation, registering, and cross-referencing of the component. This short name reflects the class and family to which the component belongs and the component number within the family.

63 A *hierarchical-to* list. A list of other components that this component is hierarchical to and for which this component can be used to satisfy dependencies to the listed components.

#### 2.1.3.2 Functional elements

64 A set of elements is provided for each component. Each element is individually defined and is self-contained.

65 A functional element is a security functional requirement that if further divided would not yield a meaningful evaluation result. It is the smallest security functional requirement identified and recognised in the CC.

66 When building PPs/STs, it is not permitted to select only one or more elements from a component. The complete set of elements of a component must be selected for inclusion in a PP/ST.

67 A unique short form of the functional element name is provided. For example the requirement name FDP_IFF.4.2 reads as follows: F - functional requirement, DP - class “User Data Protection”, _IFM - family “Information Flow Control Functions”, .4 - 4th component named “Specific Information Flow Limitation”, .2 - 2nd element of the component.

#### 2.1.3.3 Dependencies

68 For each functional component there exists a complete list of dependencies to other functional and assurance components. “No dependencies” is an acceptable list.

69 Dependencies among functional components arise when a component is not self sufficient and relies upon the functionality of, or interaction with, another component.

70 The dependency list identifies the minimum functional or assurance components needed to satisfy the security requirements associated with this component. Components which are hierarchical to the identified component may also be used to satisfy the dependency with the risk of introducing additional potential vulnerabilities.

### 2.1.4 Permitted functional component operations

71 The functional components used in the definition of the requirements in a PP, an ST, or a functional package may be exactly as specified in Chapter 2 of this part, or they may be tailored to meet a specific security objective. However, selecting and tailoring these functional components is complicated by the fact that identified component dependencies shall be considered. Thus, this tailoring is restricted to an approved set of operations.

72 A list of permitted operations is included with each functional component. Not all operations are permitted on all functional components.

73 The permitted operations are selected from the following set:

- assignment: allows the specification of an identified parameter,
- selection: allows the specification of one or more elements from a list,
- refinement: allows the addition of details.

#### 2.1.4.1 Assignment

74 Some functional component elements contain parameters or variables that enable the PP/ST author to specify a policy or a set of values for incorporation into the PP or ST to meet a specific security objective. These elements clearly identify each parameter and constraint on values that may be assigned to that parameter.

75 Any aspect of an element whose acceptable values can be unambiguously described or enumerated can be represented by a parameter. The description or enumeration limits the permissible values in such a way that all possible choices will have the same dependencies (i.e., no choice will cause the listed dependencies to change).

76 The parameter may be an attribute or rule that narrows the requirement to a specific value or range of values. For instance, based on a specified security objective, the functional component element may state that a given operation should be performed a number of times. In this case, the assignment would provide the number, or range of numbers, to be used in the parameter.

#### 2.1.4.2 Selection

77 This is the operation of picking one or more items from a list in order to narrow the scope of a component element. Each choice should be based upon the same element but give options for particularising that single requirement.

#### 2.1.4.3 Refinement

78 For all functional component elements the PP/ST author is permitted to limit the set of acceptable implementations by specifying additional detail in order to meet a security objective. Refinement of an element consists of adding these technical details.

79 The refinement does not levy any new requirements, but applies an elaboration, interpretation, or a special meaning to a requirement, rule, constant, or condition based on security objectives. The refinement shall only further restrict the set of possible acceptable functions or mechanisms to implement the requirements, but never increase it. Because refinement does not allow new requirements to be created or existing requirements to be deleted, refinement does not have any impact on the list of dependencies associated with a component.

## 2.2 Component catalogue

80 The grouping of the components in this section does not reflect any formal taxonomy.

81 Part 2 of the CC contains classes of families and components, which are rough groupings on the basis of related function or purpose, presented in alphabetic order. At the start of each class is an informative diagram that indicates the taxonomy of each class, indicating the families in each class and the components in each family. The diagram is a useful indicator of the hierarchical relationship that may exist between components.

82 In the description of the functional components, a section identifies the dependencies between this component and any other components. These dependencies reflect a normative aspect for the satisfaction of the behaviour of this component. If a component is selected in a PP, ST, or functional package the dependencies of this component shall be satisfied in order to fulfil its intended functions.

83 In Figure 2.4 the class as shown contains three families. The first family, Family 1, contains three hierarchical components, where component 2 and component 3 can

both be used to meet dependencies on component 1. Component 3 is hierarchical to component 2 and can also be used to meet dependencies on component 2.

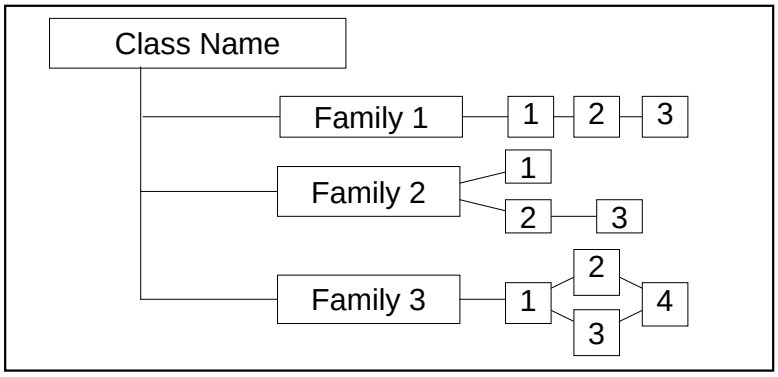



**Figure 2.4 - Sample class decomposition diagram**

84 In Family 2, there are three components; but they are not all hierarchical. Components 1 and 2 are hierarchical to no other components. Component 3 is hierarchical to component 2 and can be used to meet dependencies on component 2, but not to meet dependencies on component 1.

85 In Family 3, components 2, 3, and 4 are hierarchical to component 1. Components 2 and 3 are both hierarchical to component 1, but non-comparable. Component 4 is hierarchical to both component 2 and component 3.

86 These diagrams are meant to complement the text of the families and make identifying the relationships easier. They do not replace the “Hierarchical to:” note in each component which is the mandatory claim of hierarchy for each component.

### 2.2.1 Component changes highlighting

87 The relationship between components within a family is highlighted using a **bolding** convention. This bolding convention calls for the bolding of all new requirements. For hierarchical components, requirements and/or dependencies are bolded when they are enhanced or modified beyond the requirements of the previous component. In addition, any new or enhanced threats, application notes, and/or permitted operations beyond the previous component are also highlighted using **bold** type, whether it is in the main body or the Annexes of Part 2.

# Class FAU

# Security Audit

88 Security auditing involves recognising, recording, storing, and analysing information related to security relevant activities (i.e., any activity controlled by the TSP). The resulting audit records can be examined to determine which security relevant activities took place and who (which user) is responsible for them.

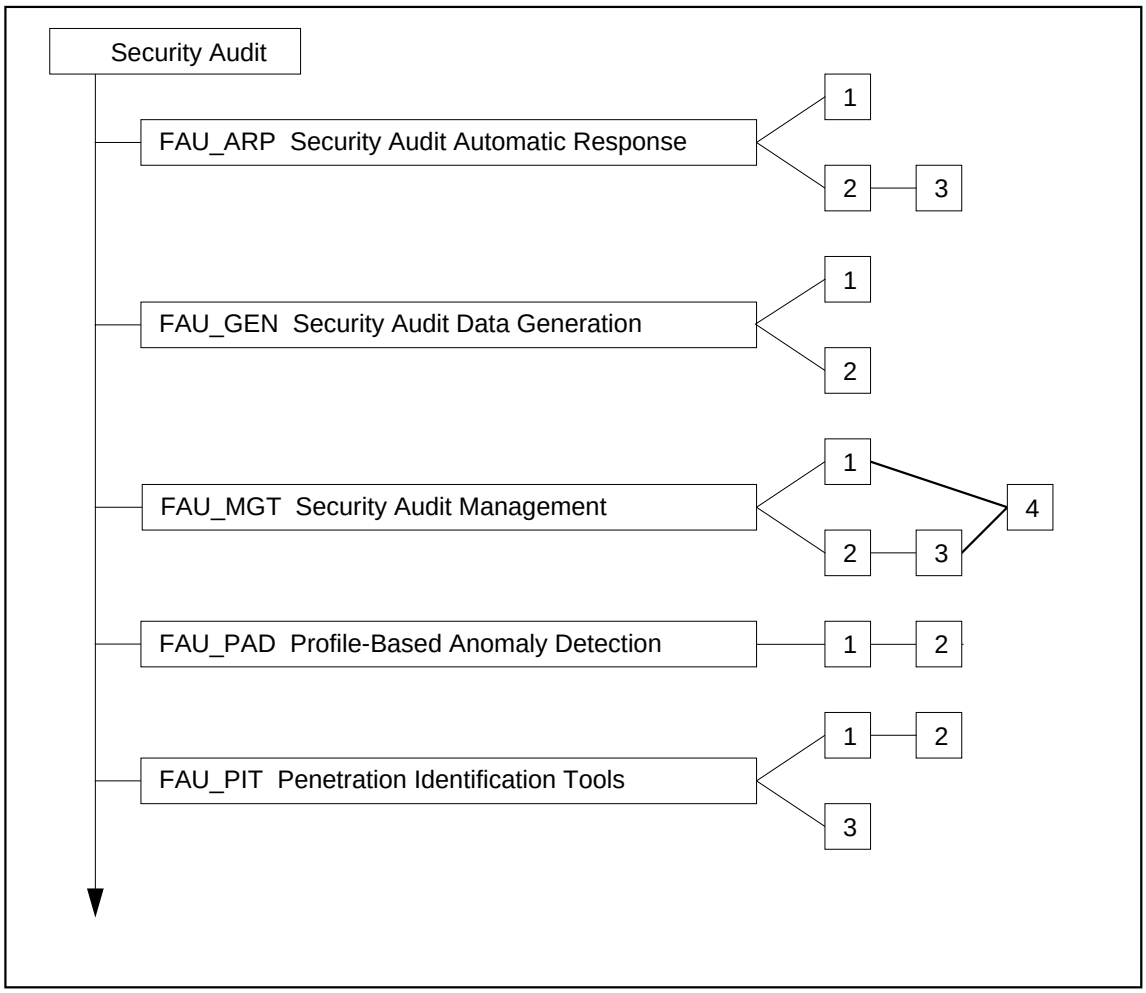



**Figure 2.5 - Security Audit Class decomposition**

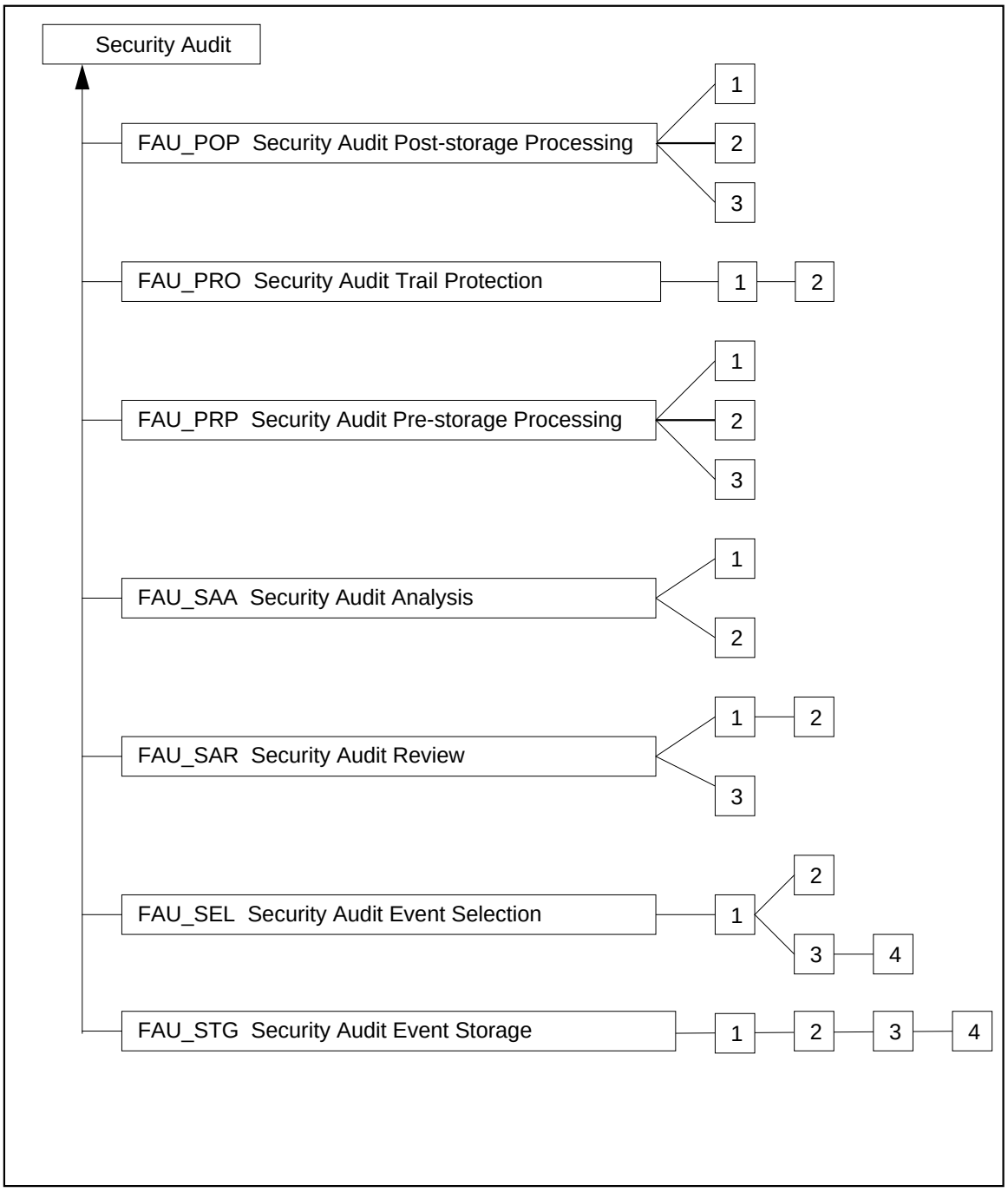



**Figure 2.6 - Security Audit Class decomposition (Cont.)**

# FAU_ARP Security Audit Automatic Response

## Family behaviour

89 This family defines the requirements specifying the conditions under which the TSF, after detection of events indicative of an imminent security violation, shall automatically react, and the reaction to those conditions that should be taken by the TSF.

## Component levelling


90 At FAU_ARP.1 Security Alarms, the TSF needs only to be passive against the imminent security violation, and have the ability to warn the authorised administrator.

91 At FAU_ARP.2 Automatic Response, the TSF shall take an active role in terminating the security violation behaviour.

92 At FAU_ARP.3 Configurable Automatic Response, the TSF shall provide the ability for the authorised administrator to choose least disruptive action from a list.

Audit : for FAU_ARP.1

93 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: Generation of an alarm to the administrator when a security violation appears imminent.

Audit : for FAU_ARP.2

94 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: Successful application of the least disruptive action that should be taken when a security violation appears imminent.

Audit : for FAU_ARP.3

95 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: Successful application of the least disruptive action that should be taken when a security violation appears imminent; and

b) Minimal: Any changes of the least disruptive actions to be taken.

### FAU_ARP.1 Security Alarms

Hierarchical to: no other components.

**FAU_ARP.1.1** **The TSF shall immediately generate an alarm to the authorised administrator upon detection of events deemed to indicate a possible security violation.**

Dependencies : **[FAU_SAA.1 Imminent Violation Analysis, or**
**FAU_PAD.1 Profile Based Anomaly Detection, or**
**FAU_PIT.1 Simple Attack Heuristics]**
**FPT_TSA.1 Basic Security Administration**

### FAU_ARP.2 Automatic Response

Hierarchical to: no other components.

**FAU_ARP.2.1** **The TSF shall take [assignment: *the least disruptive actions*] to terminate the occurrence of security relevant events upon detection of a possible security violation.**

Dependencies : **[FAU_SAA.1 Imminent Violation Analysis, or**
**FAU_PAD.1 Profile Based Anomaly Detection, or**
**FAU_PIT.1 Simple Attack Heuristics]**

### FAU_ARP.3 Configurable Automatic Response

Hierarchical to: FAU_ARP.2

**FAU_ARP.3.1** The TSF shall **be able to** take [**assignment: *list of the least disruptive actions***] to terminate the occurrence of security relevant events upon detection of a possible security violation.

**FAU_ARP.3.2** **The TSF shall provide the authorised administrator with the capability to select, from the list, the actions to be taken.**

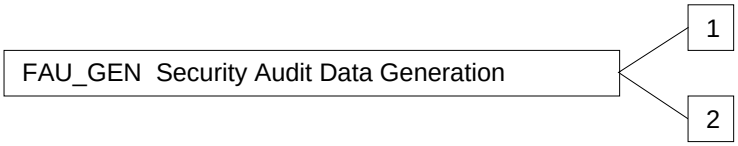
Dependencies : [FAU_SAA.1 Imminent Violation Analysis, or
FAU_PAD.1 Profile Based Anomaly Detection, or
FAU_PIT.1 Simple Attack Heuristics]
**FPT_TSA.1 Basic Security Administration**

# FAU_GEN Security Audit Data Generation

## Family behaviour

96 This family defines requirements for recording the occurrence of security relevant events that take place under TSF control. This family identifies the level of auditing, enumerates the types of events that shall be auditable by the TSF, and identifies the minimum set of audit-related information that should be provided within various audit record types.

## Component levelling




97 FAU_GEN.1 Audit Data Generation defines the level of auditable events, and specifies the list of data that shall be recorded in each record.

98 At FAU_GEN.2 User Identity Generation, the TSF shall associate auditable events to individual user identities.

## FAU_GEN.1 Audit Data Generation

Hierarchical to: no other components.

**FAU_GEN.1.1 The TSF shall be able to generate an audit record of the following auditable events:**

a) **Start-up and shutdown of the audit functions;**

b) **All auditable events for the [selection: *minimum, basic, detailed*] level of audit as defined in all functional components included in the PP/ST; and**

c) **Based on all functional components included in the PP/ST, [assignment: *other auditable events*].**

**FAU_GEN.1.2 The TSF shall record within each audit record at least the following information:**

**a) Date and time of the event, type of event, subject identity, and [selection: *success, failure*] of the event; and**

b) **For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, [assignment: *other audit relevant information*]**

Dependencies : **FIA_UID.1 Basic User Identification**

### FAU_GEN.2 User Identity Generation

Hierarchical to: no other components.

**FAU_GEN.2.1 The TSF shall be able to associate any auditable event with the identity of the user that caused the event.**

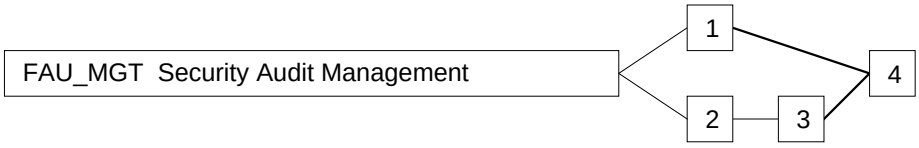
Dependencies : **FAU_GEN.1 Audit Data Generation**
**FIA_UID.1 Basic User Identification**

# FAU_MGT Security Audit Management

## Family behaviour

99 This family defines requirements that pertain to creation, destruction, or emptying of the audit trail by the authorised administrator.

## Component levelling




100 At FAU_MGT.1 Audit Trail Management, the TSF shall provide the authorised administrator with the capability to manage the audit trail.

101 At FAU_MGT.2 Audit Trail Saturation Control, the TSF shall notify the authorised administrator in case of audit trail saturation.

102 At FAU_MGT.3 Audit Trail Saturation Management, the authorised administrator shall be able to define limits to control audit trail saturation.

103 At FAU_MGT.4 Runtime Management, the authorised administrator shall be able to manage the audit trail at the runtime of the TOE.

Audit : for FAU_MGT.1

104 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: Unsuccessful operations on the audit trail.

b) Basic: Any attempt to perform an operation on the audit trail.

Audit : for FAU_MGT.2

105 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: Notification of the authorised administrator in case of audit trail saturation.

Audit : for FAU_MGT.3

106 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: Notification of the authorised administrator in case of audit trail saturation.

b) Minimal: Any changes of the pre-defined limits to control the audit trail saturation.

### FAU_MGT.1 Audit Trail Management

Hierarchical to: no other components.

**FAU_MGT.1.1** **The TSF shall provide the authorised administrator with the ability to [selection: *create, delete, empty*] the audit trail.**

Dependencies : **FAU_STG.1 Permanent Audit Trail Storage**

### FAU_MGT.2 Audit Trail Saturation Control

Hierarchical to: no other components.

**FAU_MGT.2.1** **The TSF shall generate an alarm to the authorised administrator if the size of the audit data in the audit trail exceeds a [assignment: *pre-defined limit.*]**

Dependencies : **FAU_STG.1 Permanent Audit Trail Storage**

### FAU_MGT.3 Audit Trail Saturation Management

Hierarchical to: FAU_MGT.2

**FAU_MGT.3.1** The TSF shall generate an alarm to the authorised administrator if the size of the audit data in the audit trail exceeds a pre-defined limit.

**FAU_MGT.3.2** **The TSF shall provide the authorised administrator with the ability to specify the pre-defined limit of the audit data in the audit trail at which point an alarm will be generated.**

Dependencies : No dependencies.

### FAU_MGT.4 Runtime Management

Hierarchical to: FAU_MGT.1 and FAU_MGT.3

**FAU_MGT.4.1** The TSF shall generate an alarm to the authorised administrator if the size of the audit data in the audit trail exceeds a pre-defined limit.

**FAU_MGT.4.2** **The TSF shall provide the authorised administrator with the ability to manage the audit trail at any time during the operation of the TOE.**

Dependencies : No dependencies.

# FAU_PAD Profile-Based Anomaly Detection

## Family behaviour

107 This family defines requirements for automated tools capable of constructing profiles of established system usage, and detecting new system activity that deviates from these established patterns of usage. Suspicion ratings are calculated that represent how well an individual's current activity corresponds to their established usage patterns. The more anomalous the activity, the greater the suspicion rating becomes. When a suspicion rating reaches a predefined threshold, the anomalous activity is brought to the attention of the administrator and the TSF may take steps to prevent continued activity from individuals with high suspicion ratings.

## Component levelling


108 In FAU_PAD.1 Profile Based Anomaly Detection, the TSF maintains individual *profiles* of system usage, where a profile represents the historical patterns of usage performed by members of the profile's *profile target group.* A profile target group refers to a group of one or more individuals (e.g., a single user, users who share a group ID or group account, users who operate under an assigned role, users of an entire system or network node) who interact with the TSF. Each member of a profile target group is assigned an individual *suspicion rating* that represents how well that member's current activity corresponds to the established patterns of usage represented in the profile. When the suspicion rating exceeds a predefined threshold, the administrator is notified. This analysis can be performed at runtime or during a post-collection batch-mode analysis.

109 In FAU_PAD.2 Dynamic Profile-Based Surveillance and Response, the administrator shall also be able to modify the profile metrics under which historical usage patterns are established, and the threshold conditions under which anomalous activity is reported to the administrator and terminated by the TSF.

## Audit :

110 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: Enabling and disabling of any of the analysis mechanisms;

b) Minimal: Notifications made to the authorised administrator; and

c) Minimal: Automated responses performed by the tool.

d) Basic: Any changes to the configuration of the analysis mechanism.

## FAU_PAD.1 Profile Based Anomaly Detection

Hierarchical to: no other components.

**FAU_PAD.1.1** **The TSF shall be able to maintain profiles of system usage, where an individual profile represents the historical patterns of usage performed by the member(s) of [assignment: *specify the profile target group*].**

**FAU_PAD.1.2** **The TSF shall be able to maintain a suspicion rating associated with each user whose activity is recorded in a profile, where the suspicion rating represents the degree to which the user's current activity is found inconsistent with the established patterns of usage represented in the profile.**

**FAU_PAD.1.3** **The TSF shall be able to indicate an imminent violation of the TSP when a user's suspicion rating exceeds the following threshold conditions [assignment: *conditions under which anomalous activity is reported by the TSF*].**

Dependencies : No dependencies.

## FAU_PAD.2 Dynamic Profile-Based Surveillance and Response

Hierarchical to: FAU_PAD.1

**FAU_PAD.2.1** The TSF shall be able to maintain profiles of system usage, where an individual profile represents the historical patterns of usage performed by the member(s) of [assignment: *specify the profile target group*].

**FAU_PAD.2.2** The TSF shall be able to maintain a suspicion rating associated with each individual whose activity is recorded in a profile, where the suspicion rating represents the degree to which the individual's current activity is found inconsistent with the established patterns of usage represented in the profile.

**FAU_PAD.2.3** The TSF shall be able to indicate an imminent violation of the TSP when a user's suspicion rating exceeds the following threshold conditions [assignment: *specify conditions under which anomalous activity is reported by the TSF*].

**FAU_PAD.2.4** **The TSF shall provide an interface to allow the authorised administrator to modify [assignment: *list of profile metrics subject to dynamic configuration*].**

**FAU_PAD.2.5** **The TSF shall provide an interface to allow the authorised administrator to modify the threshold conditions under which [selection: *anomalous activity is reported to the administrator, action is taken to terminate further activity from the responsible individual*].**

Dependencies : No dependencies.

# FAU_PIT Penetration Identification Tools

## Family behaviour

111 This family defines requirements for automated tools capable of analysing system activity, and comparing this activity against *signature events* (events whose occurrence are representative of known real or potential intrusive activity) and event sequences that represent entire known penetration scenarios. Such events, if detected during an analysis of system activity, warrant human review and in certain environments may warrant automated intervention. System activity may be discerned through an examination of various data (or combinations thereof) handled by the TSF, such as security audit logs, network datagrams, and resource management/accounting data.

## Component levelling


112 In FAU_PIT.1 Simple Attack Heuristics, the TSF shall be able to detect the occurrence of signature events that represent a significant threat to TSP enforcement. Such events are identified and called out to the administrator for human review. This search for signature events may occur in real-time or during a post-collection batch-mode analysis.

113 In FAU_PIT.2 Complex Attack Heuristics, the TSF shall be able to represent and detect multi-step intrusion scenarios. The TSF is able to compare system events (possibly performed by multiple individuals) against event sequences known to represent entire intrusion scenarios. The TSF shall be able to indicate when a signature event or event sequence is found to match a signature event that indicates a potential violation of the TSP.

114 In FAU_PIT.3 Dynamic Run-Time Attack Management, the TSF shall allow an authorised administrator to dynamically change the set of defined signature events and event sequences (i.e., the attack heuristics).

## Audit :

115 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: Enabling and disabling of any of the analysis mechanisms;

b) Minimal: Indications made of imminent violations of the TSP; and

c) Basic: Any changes to the configuration of the analysis mechanism.

### FAU_PIT.1 Simple Attack Heuristics

Hierarchical to: no other components.

**FAU_PIT.1.1** **The TSF shall be able to maintain an internal representation of the following signature events [assignment: *a subset of system events*] that may indicate a violation of the TSP.**

**FAU_PIT.1.2** **The TSF shall be able to compare the signature events against the record of system activity discernable from an examination of [assignment: *specify the information to be used to determine system activity*].**

**FAU_PIT.1.3** **The TSF shall be able to indicate an imminent violation of the TSP when a system event is found to match a signature event that indicates a potential violation of the TSP.**

Dependencies : No dependencies.

### FAU_PIT.2 Complex Attack Heuristics

Hierarchical to: FAU_PIT.1

**FAU_PIT.2.1** The TSF shall be able to maintain an internal representation of the following **event sequences of known intrusion scenarios [assignment: *list of sequences of system events whose occurrence are representative of known penetration scenarios*] and** the following signature events [assignment: *a subset of system events*] that may indicate a potential violation of the TSP.

**FAU_PIT.2.2** The TSF shall be able to compare the signature events **and event sequences** against the record of system activity discernable from an examination of [assignment: *specify the information to be used to determine system activity*].

**FAU_PIT.2.3** The TSF shall be able to indicate an imminent violation of the TSP when **system activity** is found to match a signature event **or event sequence** that indicates a potential violation of the TSP.

Dependencies : No dependencies.

### FAU_PIT.3 Dynamic Run-Time Attack Management

Hierarchical to: no other components

**FAU_PIT.3.1** **The TSF shall allow the authorised administrator to dynamically [selection: *add, modify, delete*] defined signature events and event sequences.**

Dependencies : **FAU_PIT.1 Simple Attack Heuristics**

# FAU_POP Security Audit Post-storage Processing

Family behaviour

116 This family defines requirements on the TSF for the capability to transform the permanent representation of a single security audit event, stored in the audit trail, into a useful and consistent format for its subsequent attempted use.

Component levelling


117 FAU_POP.1 Human Understandable Format provides the capability to present audit data in a form understandable (e.g., for review) by a human user.

118 FAU_POP.2 Automated Treatment Format provides the capability to present audit data in a useful format for automated treatment (e.g., transfer, selection, analysis), by the TSF itself or any machine user.

119 FAU_POP.3 Flexible Format provides the capability to rearrange audit data.

Audit : for FAU_POP.1 and FAU_POP.2

120 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: Any specific operation performed to process audit data stored in the audit trail.

Audit : For FAU_POP.3

121 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: Any specific operation performed to restructure or reorder audit data stored in the audit trail.

## FAU_POP.1 Human Understandable Format

Hierarchical to: no other components.

**FAU_POP.1.1 The TSF shall be able to generate a human understandable presentation of any audit data stored in the permanent audit trail.**

Dependencies : **FAU_STG.1 Permanent Audit Trail Storage**

### FAU_POP.2 Automated Treatment Format

Hierarchical to: no other components.

FAU_POP.2.1 **The TSF shall be able to generate for automated treatment a valid representation of any audit data stored in the permanent audit trail.**

Dependencies : **FAU_STG.1 Permanent Audit Trail Storage**

### FAU_POP.3 Flexible Format

Hierarchical to: no other components.

FAU_POP.3.1 **The TSF shall be able to generate a new representation of the audit trail, by rearranging the order and content of the audit data.**

Dependencies : **FAU_STG.1 Permanent Audit Trail Storage**

# FAU_PRO Security Audit Trail Protection

Family behaviour

122 This family provides requirements to protect the audit trail from unauthorised modification, disclosure, or destruction. This pertains to periods where the audit trail is exchanged between, or processed or stored within parts of a TOE.

Component levelling


123 At FAU_PRO.1 Restricted Audit Trail Access, access to the audit trail is restricted to the authorised administrator.

124 At FAU_PRO.2 Extended Audit Trail Access, full access to the audit trail is restricted only to the authorised administrator, and authorised users could be granted read access.

Audit :

125 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: Successful requests to read, modify or destroy the audit trail.

b) Basic: Any attempt to read, modify or destroy the audit trail.

## FAU_PRO.1 Restricted Audit Trail Access

Hierarchical to: no other components.

**FAU_PRO.1.1** **The TSF shall restrict access to the audit trail to the authorised administrator.**

Dependencies : **FAU_STG.1 Permanent Audit Trail Storage**
**FPT_TSA.1 Basic Security Administration**

## FAU_PRO.2 Extended Audit Trail Access

Hierarchical to: FAU_PRO.1

**FAU_PRO.2.1** The TSF shall restrict **full** access to the audit trail to the authorised administrator.

**FAU_PRO.2.2** **The TSF shall provide only authorised users with the capability to read [assignment: *list of audit information*] from the audit trail.**

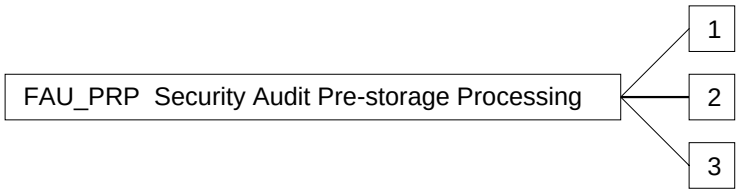
Dependencies : FAU_STG.1 Permanent Audit Trail Storage
FPT_TSA.1 Basic Security Administration

# FAU_PRP Security Audit Pre-storage Processing

## Family behaviour

126 This family defines requirements on the TSF for the capability to transform the generated representation of a single security audit event into a useful and consistent format for its subsequent attempted use. Audit data should be placed into a useful format for delivery to authorised users, or processes acting on their behalf.

## Component levelling




127 FAU_PRP.1 Human Understandable Format provides the capability to present audit data understandable (e.g., for review) by a human user,

128 FAU_PRP.2 Automated Treatment Format provides the capability to present audit data in a useful format for automated treatment (e.g., transfer, selection, analysis), by the TSF itself or any other machine user.

129 FAU_PRP.3 Flexible Format provides the capability to rearrange audit data.

## Audit :

130 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: Any specific operation performed to process audit data.

### FAU_PRP.1 Human Understandable Format

Hierarchical to: no other components.

**FAU_PRP.1.1 The TSF shall be able to generate a human understandable presentation of any audit data generated.**

Dependencies : **FAU_GEN.1 Audit Data Generation**

### FAU_PRP.2 Automated Treatment Format

Hierarchical to: no other components.

FAU_PRP.2.1 **The TSF shall be able to generate for automated treatment a valid representation of any audit data generated.**

Dependencies : **FAU_GEN.1 Audit Data Generation**

### FAU_PRP.3 Flexible Format

Hierarchical to: no other components.

FAU_PRP.3.1 **The TSF shall be able to generate new audit data, by rearranging the order and content of the audit data generated.**

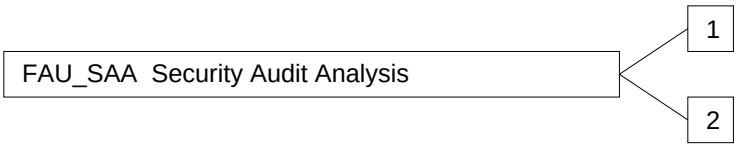
Dependencies : **FPT_TSA.1 Basic Security Administration**

# FAU_SAA Security Audit Analysis

Family behaviour

131 This family defines requirements for automated means which analyse system activity and audit data looking for possible or real security violations. This analysis may work in support of intrusion detection, or automatic response to an imminent security violation.

Component levelling




132 In FAU_SAA.1 Imminent Violation Analysis, basic threshold detection on the basis of a fixed rule set is required.

133 In FAU_SAA.2 Configurable Violation Analysis, the rule set shall be modifiable by the authorised administrator.

Audit : for FAU_SAA.1

134 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: detection of imminent violation by the analysis.

Audit : for FAU_SAA.2

135 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: Any changes to the configuration of the analysis functions by an authorised administrator.

## FAU_SAA.1 Imminent Violation Analysis

Hierarchical to: no other components.

**FAU_SAA.1.1 The TSF shall be able to apply a set of rules in monitoring the audited events and based upon these rules indicate a potential violation of the TSP.**

**FAU_SAA.1.2 The set of rules shall be:**

a) **Accumulation or combination of [assignment: *subset of defined auditable events*] known to indicate a possible or imminent security violation;**

b) **[assignment: *any other rules*].**

Dependencies : **FAU_GEN.1 Audit Data Generation**

### FAU_SAA.2 Configurable Violation Analysis

Hierarchical to: no other components.

**FAU_SAA.2.1** **The TSF shall restrict to the authorised administrator the [selection: *addition, modification, deletion*] of rules from the set of rules.**

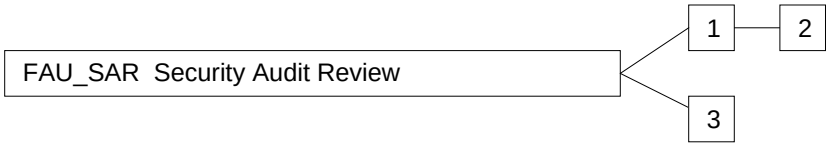
Dependencies : **FAU_SAA.1 Imminent Violation Analysis**
**FPT_TSA.1 Basic Security Administration**

# FAU_SAR Security Audit Review

Family behaviour

136 This family defines the requirements for audit tools that should be available to authorised users to assist in the review of audit data.

Component levelling




137 At FAU_SAR.1 Restricted Audit Review, only the authorised administrator is allowed to use the review tools,

138 At FAU_SAR.2 Extended Audit Review, authorised users are allowed to a limited use of the review tools.

139 FAU_SAR.3 Selectable Audit Review requires audit review tools to select the audit data to be reviewed based on multiple criteria.

## FAU_SAR.1 Restricted Audit Review

Hierarchical to: no other components.

**FAU_SAR.1.1** **The TSF shall provide audit review tools, with the ability to view the audit data.**

**FAU_SAR.1.2** **The TSF shall restrict use of the audit review tools to the authorised administrator.**

Dependencies : **FAU_STG.1 Permanent Audit Trail Storage**
**FPT_TSA.1 Basic Security Administration**
**FAU_PRO.1 Restricted Audit Trail Access**

## FAU_SAR.2 Extended Audit Review

Hierarchical to: FAU_SAR.1

**FAU_SAR.2.1** The TSF shall provide audit review tools, with the ability to view the audit data.

**FAU_SAR.2.2** The TSF shall restrict **full** use of the audit review tools to the authorised administrator.

**FAU_SAR.2.3** **The TSF shall provide only authorised users with limited use of the audit review tools.**

Dependencies : FAU_STG.1 Permanent Audit Trail Storage
FPT_TSA.1 Basic Security Administration
**FAU_PRO.2 Extended Audit Trail Access**

### FAU_SAR.3 Selectable Audit Review

Hierarchical to: no other components.

**FAU_SAR.3.1** **The TSF shall provide audit review tools with the ability to perform searches and sorting of audit data based on [assignment: *multiple criteria with logical relations*].**

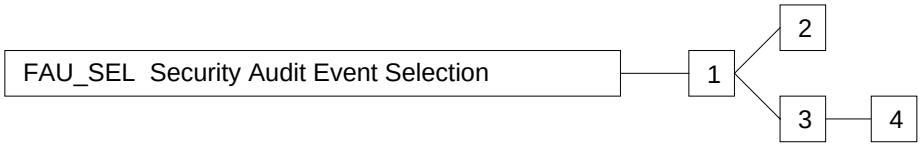
Dependencies : **FAU_SAR.1 Restricted Audit Review**

# FAU_SEL Security Audit Event Selection

Family behaviour

140 This family defines requirements to select the events to be audited during TOE operation. It defines requirements to include or exclude events from the set of auditable events.

Component levelling




141 FAU_SEL.1 Selective Audit, requires the ability to include or exclude events from the set of audited events based upon attributes to be specified by the PP/ST author.

142 FAU_SEL.2 Runtime Selection Mode requires runtime configurability of the audit functions.

143 At FAU_SEL.3 Restricted Runtime Display Mode only the authorised administrator is allowed to display the auditable event selection criteria.

144 At FAU_SEL.4 Extended Runtime Display Mode authorised users are allowed to display auditable event selection criteria.

Audit :

145 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: All modifications to the audit configuration that occur while the audit collection functions are operating.

## FAU_SEL.1 Selective Audit

Hierarchical to: no other components.

**FAU_SEL.1.1 The TSF shall be able to include or exclude auditable events from the set of audited events based on the following attributes:**

a) **[selection: *Object identity, User identity, Subject identity, Host identity, Event Type*]**

**b) [assignment: *list of additional attributes*] that audit selectivity is based upon.**

Dependencies : **FAU_GEN.1 Audit Data Generation**

### FAU_SEL.2 Runtime Selection Mode

Hierarchical to: FAU_SEL.1

**FAU_SEL.2.1** The TSF shall be able to include or exclude auditable events from the set of audited events based on the following attributes:

a) [selection: *Object identity, User identity, Subject identity, Host identity, Event Type*]

b) [assignment: *list of additional attributes*] that audit selectivity is based upon.

**FAU_SEL.2.2** **The TSF shall provide the authorised administrator with the capability to select, at any time during the operation of the TOE, which events are to be audited.**

Dependencies : **FPT_TSA.1 Basic Security Administration**

### FAU_SEL.3 Restricted Runtime Display Mode

Hierarchical to: FAU_SEL.1

**FAU_SEL.3.1** The TSF shall be able to include or exclude auditable events from the set of audited events based on the following attributes:

a) [selection: *Object identity, User identity, Subject identity, Host identity, Event Type*]

b) [assignment: *list of additional attributes*] that audit selectivity is based upon.

**FAU_SEL.3.2** **The TSF shall restrict to the authorised administrator the capability to display, at any time during the operation of the TOE, which events are being audited.**

Dependencies : **FPT_TSA.1 Basic Security Administration**

### FAU_SEL.4 Extended Runtime Display Mode

Hierarchical to: FAU_SEL.3

**FAU_SEL.4.1** The TSF shall be able to include or exclude auditable events from the set of audited events based on the following attributes:

a) [selection: *Object identity, User identity, Subject identity, Host identity, Event Type*]

b) [assignment: *list of additional attributes*] that audit selectivity is based upon.

**FAU_SEL.4.2** The TSF shall restrict to the authorised administrator the **full** capability to display, at any time during the operation of the TOE, which events are being audited.

**FAU_SEL.4.3** **The TSF shall provide only authorised users with the capability to display, at any time during the operation of the TOE, which events are being audited.**

Dependencies : FPT_TSA.1 Basic Security Administration

# FAU_STG Security Audit Event Storage

Family behaviour

146 This family defines the requirement that the TSF be able to create a permanent audit trail of security audit events for later use. This family also addresses the control of the loss of audit information.

Component levelling


147 At FAU_STG.1 Permanent Audit Trail Storage, a permanent audit trail is created.

148 At FAU_STG.2 Enumeration of Audit Data Loss, conditions under which audit data loss occurs due to system failure shall be enumerated by the developer.

149 FAU_STG.3 Prevention of Audit Data Loss adds the requirement that the TSF be capable of preventing audit data loss due to exhaustion of storage space.

150 FAU_STG.4 Manageable Prevention of Audit Data Loss adds the requirement that the authorised administrator shall define the action the TSF shall take for preventing audit data loss due to exhaustion of storage space.

## FAU_STG.1 Permanent Audit Trail Storage

Hierarchical to: no other components.

**FAU_STG.1.1** **The TSF shall store generated audit records in a permanent audit trail.**

Dependencies : **FAU_GEN.1 Audit Data Generation**

## FAU_STG.2 Enumeration of Audit Data Loss

Hierarchical to: FAU_STG.1

**FAU_STG.2.1** The TSF shall store generated audit records in a permanent audit trail.

**FAU_STG.2.2** **The TSF shall limit the number of audit records lost due to system [selection: *audit storage exhaustion, failure, attack*].**

Dependencies : **FAU_GEN.1 Audit Data Generation**

### FAU_STG.3 Prevention of Audit Data Loss

Hierarchical to: FAU_STG.2

**FAU_STG.3.1** The TSF shall store generated audit records in a permanent audit trail.

**FAU_STG.3.2** The TSF shall limit the number of audit records lost due to system [selection: *audit storage exhaustion, failure, attack*].

**FAU_STG.3.3** **In the event of audit storage exhaustion, the TSF shall be capable of [selection: *ignoring, preventing*] the occurrence of auditable actions, except those taken by the authorised administrator.**

Dependencies : FAU_GEN.1 Audit Data Generation

### FAU_STG.4 Manageable Prevention of Audit Data Loss

Hierarchical to: FAU_STG.3

**FAU_STG.4.1** The TSF shall store generated audit records in a permanent audit trail.

**FAU_STG.4.2** The TSF shall limit the number of audit records lost due to system [selection: *audit storage exhaustion, failure, attack*].

**FAU_STG.4.3** **The TSF shall provide to the authorised administrator the capability to define if the TSF shall ignore or prevent,** in the event of audit storage exhaustion, the occurrence of auditable actions, except those taken by the authorised administrator.

Dependencies : FAU_GEN.1 Audit Data Generation

# Class FCO

# Communication

151 This class provides two families specifically concerned with assuring the identity of a party participating in a data exchange. These families are related to assuring the identity of the originator of transmitted information (proof of origin) and assuring the identity of the recipient of transmitted information (proof of receipt). These families ensure that an originator cannot deny having sent the message, nor can the recipient deny receiving it.

152 Figure 2.7 shows the decomposition of this class into its consituent components.

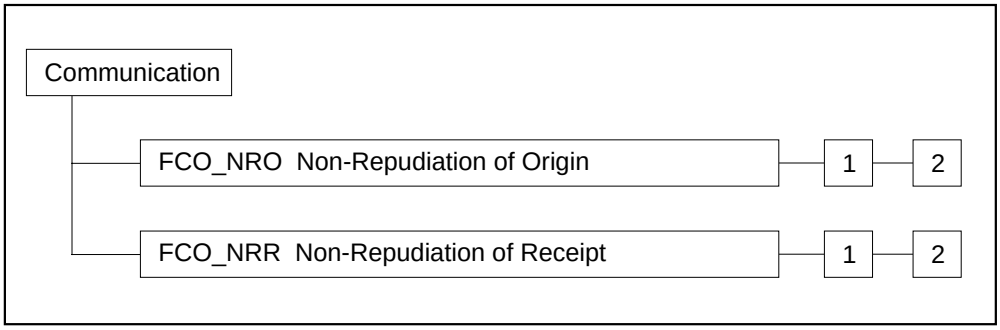


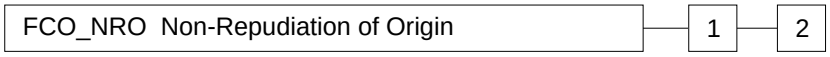

**Figure 2.7 - Communication class decomposition**

# FCO_NRO Non-Repudiation of Origin

## Family behaviour

153 Non-repudiation of origin ensures that the originator of information cannot successfully deny sending the information. The TSF shall provide a method to ensure that a subject that receives information during a data exchange is provided with evidence of the origin of the information. This evidence can be verified by either this subject or other subjects.

## Component levelling




154 FCO_NRO.1 Enforced Proof of Origin requires that the TSF always generate evidence of origin for transmitted information.

155 FCO_NRO.2 Selective Proof of Origin requires the TSF to provide subjects with the capability to request evidence of origin on information.

Audit : for FCO_NRO.1 and FCO_NRO.2

156 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: The invocation of the non-repudiation service.

b) Basic: Identification of the information, the destination, and a copy of the evidence provided.

c) Basic: The identity of the user which requested a verification of the evidence.

d) Detailed: The user data content present in the information.

Audit : for FCO_NRO.2

157 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: The identity of the user which requested that evidence of origin would be generated.

## FCO_NRO.1 Enforced Proof of Origin

Hierarchical to: no other components.

**FCO_NRO.1.1** **The TSF shall generate evidence of origin for transmitted [assignment: *list of information types*].**

**FCO_NRO.1.2** **The TSF shall be able to relate the [assignment: *list of attributes*] of the originator of the information, and the [assignment: *list of information fields*] of the information to which the evidence applies.**

**FCO_NRO.1.3** **The TSF shall provide a capability to verify the evidence of origin of information to [selection: *originator, recipient,* [assignment: *list of third parties*]] given [assignment: *limitations on the evidence of origin*].**

Dependencies : **FIA_UID.1 Basic User Identification**

## FCO_NRO.2 Selective Proof of Origin

Hierarchical to: FCO_NRO.1

**FCO_NRO.2.1** The TSF shall **be able to** generate evidence of origin for transmitted [assignment: *information*].

**FCO_NRO.2.2** The TSF shall be able to relate the [assignment: *list of attributes*] of the originator of the information, and the [assignment: *list of information fields*] of the information to which the evidence applies.

**FCO_NRO.2.3** The TSF shall provide a capability to verify the evidence of origin of information to [selection: *originator, recipient,* [assignment: *list of third parties*]] given [assignment: *limitations on the evidence of origin*].

**FCO_NRO.2.4** **The TSF shall provide the [selection: *originator, recipient,* [assignment: *list of third parties*]] with the ability to request evidence of origin on transmitted information.**

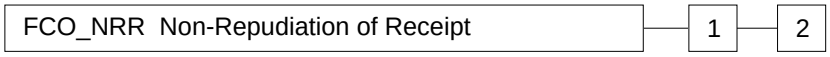
Dependencies : FIA_UID.1 Basic User Identification

# FCO_NRR Non-Repudiation of Receipt

## Family behaviour

158 Non-repudiation of receipt ensures that the recipient of information cannot successfully deny receiving the information. The TSF shall provide a method to ensure that a subject that transmits information during a data exchange is provided with evidence of receipt of the information. This evidence can be verified by either this subject or other subjects.

## Component levelling




159 FCO_NRR.1 Enforced Proof of Receipt requires that the TSF always generate evidence of receipt for received information.

160 FCO_NRR.2 Selective Proof of Receipt requires the TSF to provide subjects with a capability to request evidence of receipt on information.

Audit : for FCO_NRR.1 and FCO_NRR.2

161 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: The invocation of the non-repudiation service.

b) Basic: Identification of the information, the destination, and a copy of the evidence provided.

c) Basic: The identity of the user which requested a verification of the evidence.

d) Detailed: The user data content present in the information.

Audit : for FCO_NRO.2

162 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: The identity of the user which requested that evidence of origin would be generated.

## FCO_NRR.1 Enforced Proof of Receipt

Hierarchical to: no other components.

**FCO_NRR.1.1** **The TSF shall generate evidence of receipt for received [assignment: *information*].**

**FCO_NRR.1.2** **The TSF shall be able to relate the [assignment: *list of attributes*] of the recipient of the information, and the [assignment: *list of information fields*] of the information to which the evidence applies.**

**FCO_NRR.1.3** **The TSF shall provide a capability to verify the evidence of receipt of information to [selection: *originator, recipient,* [assignment: *list of third parties*]] given [assignment: *limitations on the evidence of receipt*].**

Dependencies : **FIA_UID.1 Basic User Identification**

## FCO_NRR.2 Selective Proof of Receipt

Hierarchical to: FCO_NRR.1

**FCO_NRR.2.1** The TSF shall **be able to** generate evidence of receipt for received [assignment: *information*].

**FCO_NRR.2.2** The TSF shall be able to relate the [assignment: *list of attributes*] of the recipient of the information, and the [assignment: *list of information fields*] of the information to which the evidence applies.

**FCO_NRR.2.3** The TSF shall provide a capability to verify the evidence of receipt of information to [selection: *originator, recipient,* [assignment: *list of third parties*]] given [assignment: *limitations on the evidence of receipt*].

**FCO_NRR.2.4** **The TSF shall provide the [selection: *originator, recipient,* [assignment: *list of third parties*]] the ability to request evidence of receipt on information.**

Dependencies : FIA_UID.1 Basic User Identification

# Class FDP

# User Data Protection

163 This class contains families specifying requirements for TOE security functions and TOE security function policies related to protecting user data. FDP is split into five groups of families (listed below) which address user data within a TOE, during import, export, and storage as well as security attributes directly related to user data.

164 The families in this class are organised into five groups:

a) Forms of Data Protection:

- FDP_ACC Access Control Policy; and
- FDP_IFC Information Flow Control Policy.

Components in these families permit the PP/ST author to specify data protection security function policies to address the security objectives.

b) Data Protection Security Function Policies:

- FDP_ACF Access Control Functions;
- FDP_IFF Information Flow Control Functions;
- FDP_ITT Internal TOE Transfer;
- FDP_SDI Stored Data Integrity; and
- FDP_ROL Rollback.

Components in these families address functions that protect information and objects through enforcement of the data protection.

c) Security Attribute Management:

- FDP_ACI Object Attributes Initialisation;
- FDP_SAM Security Attribute Modification; and
- FDP_SAQ Security Attribute Query.

Components in these families address managing the security attributes related to user data.

d) Off-line Storage and Communication:

- FDP_ETC Export to Outside TSF Control; and
- FDP_ITC Import from Outside TSF Control.

Components in these families address the trustworthy transfer into or out of the TSC.

e) Inter-TSF Communication:

- FDP_UCT Inter-TSF User Data Confidentiality Transfer Protection; and

- FDP_UIT Inter-TSF User Data Integrity Transfer Protection.

Components in these families address communication between TSFs. The two end points are assumed to be known and to some degree pre-configured to support the communication.

Figures 2.8 and 2.9 show the decomposition of this class into its constituent components.

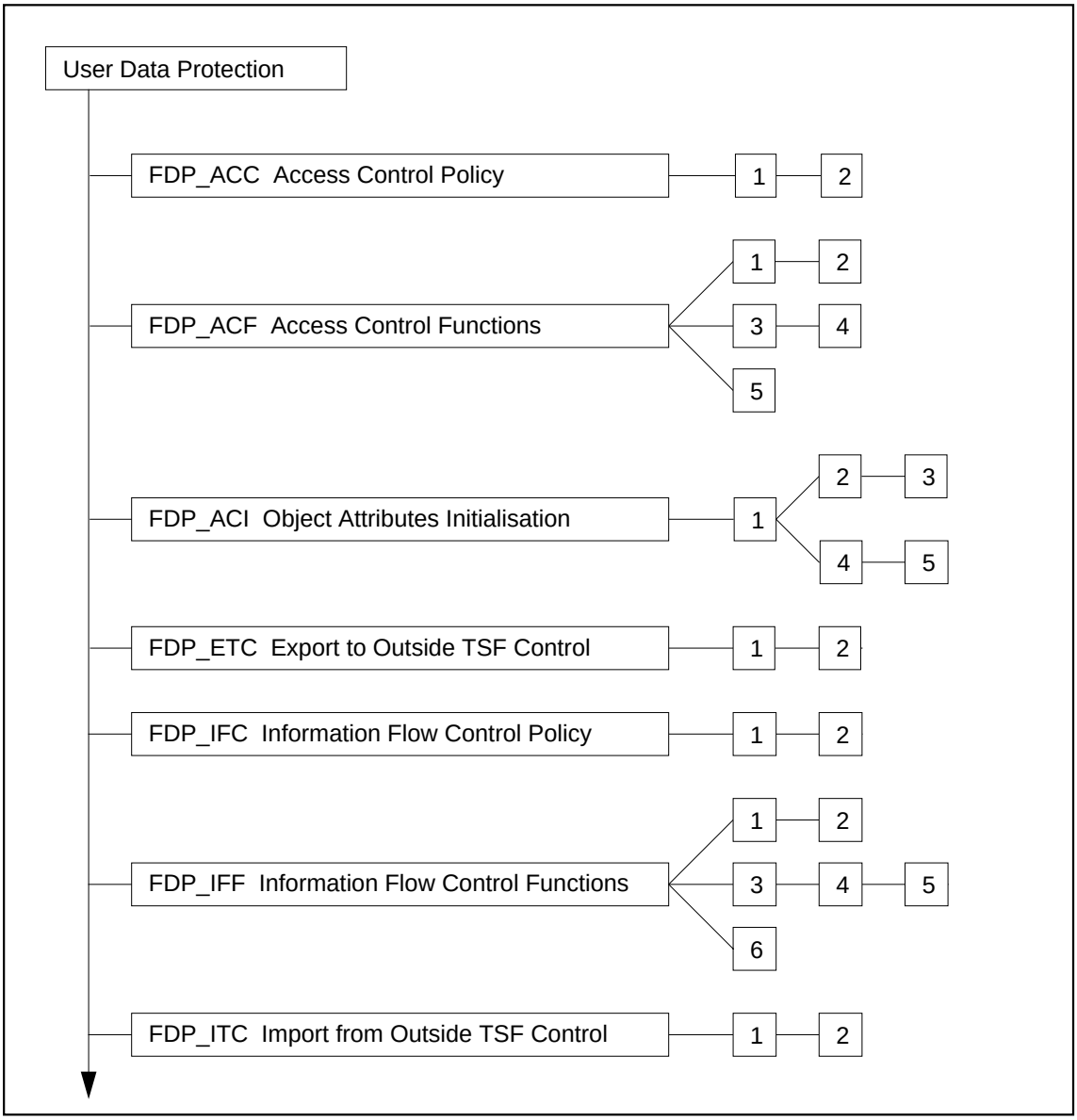



**Figure 2.8 - User Data Protection class decomposition**

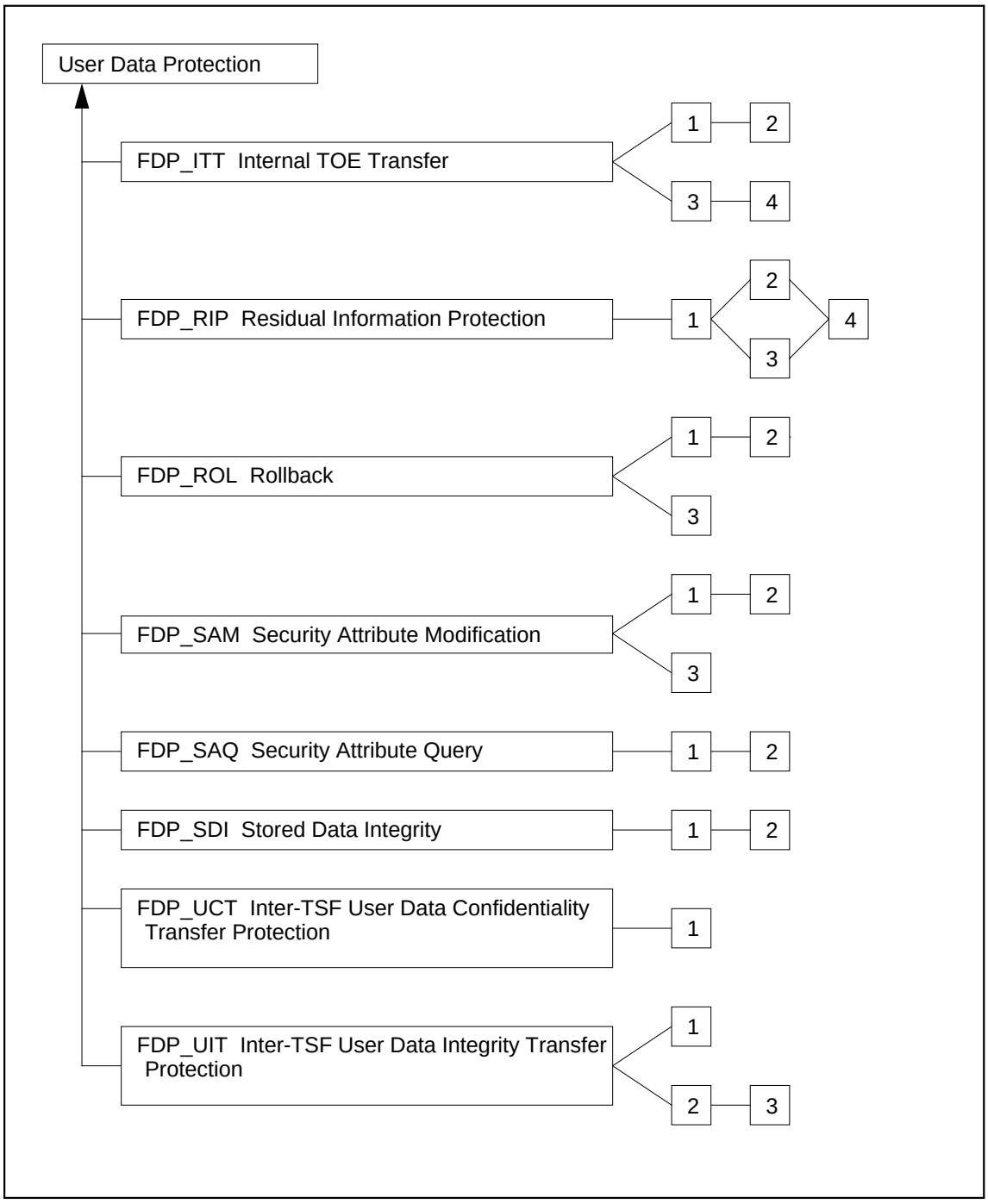


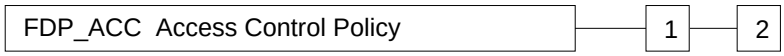

**Figure 2.9 - User Data Protection class decomposition (cont.)**

## FDP_ACC Access Control Policy

Family behaviour

165 This family defines the scope of control of the access control policies that form the access control portion of the TSP. This scope of control is characterised by three sets: the subjects under control of the policy, the objects under control of the policy, and the operations among controlled subjects and controlled objects that are covered by the policy. The criteria allows multiple policies to exist, each having a unique name.

Component levelling




166 FDP_ACC.1 Subset Object Access Control requires that one or more access control SFPs be in place for a subset of the possible operations on a subset of the objects in the TOE.

167 FDP_ACC.2 Complete Object Access Control requires that one SFP cover all operations on subjects and objects covered by the SFP.

### FDP_ACC.1 Subset Object Access Control

Hierarchical to: no other components.

**FDP_ACC.1.1** **The TSF shall enforce the [assignment: *access control SFP*] on [assignment: *list of subjects, objects, and operations among subjects and objects covered by the SFP*].**

Dependencies : FDP_ACF.1 Single Security Attribute Access Control

### FDP_ACC.2 Complete Object Access Control

Hierarchical to: FDP_ACC.1

**FDP_ACC.2.1** The TSF shall enforce the [assignment: access control SFP] on [assignment: list of subjects and objects] and **all operations among subjects and objects covered by the SFP.**

**FDP_ACC.2.2** **The TSF shall ensure that all operations between any subject in the TSC and any object within the TSC are covered by the SFP.**

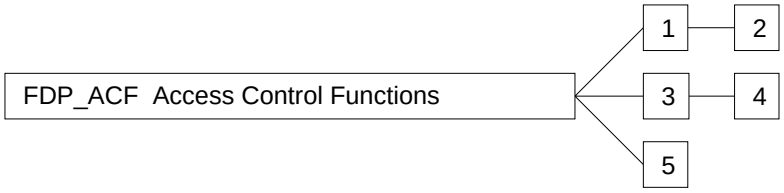
Dependencies : FDP_ACF.1 Single Security Attribute Access Control

# FDP_ACF Access Control Functions

## Family behaviour

168 This family describes specific functions that can implement the rules for access control and is to be used in conjunction with FDP_ACC which specifies the access control policies.

## Component levelling




169 The family defines three types of components addressing security attribute usage (FDP_ACF.1 and FDP_ACF.2), flexible characteristics of policies (FDP_ACF.3 and FDP_ACF.4), and fixed characteristics of policies (FDP_ACF.5). These components are to be combined to describe the function implementing the SFP as defined in FDP_ACC. The PP/ST author is also required to use some elements (for example, FDP_ACF.1.2) multiple times to address multiple policies in the TOE.

170 The FDP_ACF.1 Single Security Attribute Access Control component allows the TSF to enforce access based upon a single security attribute.

171 The FDP_ACF.2 Multiple Security Attribute Access Control component requires the TSF to enforce access control based upon multiple security attributes.

172 The FDP_ACF.3 Access Authorisation component adds the ability to grant access to the object in addition to security attribute modification.

173 The FDP_ACF.4 Access Authorisation and Denial component provides the ability to deny access to the object in addition to the requirements of FDP_ACF.3.

174 The FDP_ACF.5 Fixed Access Control component prohibits the ability to change security attributes, deny access, or grant access to the object.

## Audit : for FDP_ACF.1 and FDP_ACF.2

175 The following events shall be audited, if FAU_GEN Security Audit Data Generation is included in the PP/ST:

a) Minimal: Successful requests to perform an operation on an object covered by the SFP.

b) Basic: All requests to perform an operation on an object covered by the SFP.

c) Detailed: The specific security attributes used in making an access check.

Audit : for FDP_ACF.3 and FDP_ACF.4

176 The following events shall be audited, if FAU_GEN Security Audit Data Generation is included in the PP/ST:

a) Minimal: Successful specification of granting or denying access to an object.

b) Basic: Unsuccessful attempts to specify the granting or denying of access to an object.

c) Detailed: The identity of the user or subject who specifies, or attempts to specify, the granting or denying of access to an object.

### FDP_ACF.1 Single Security Attribute Access Control

Hierarchical to: no other components.

**FDP_ACF.1.1** **The TSF shall enforce the [assignment: *access control SFP*] to objects based on [assignment: *attribute, named group of attributes*].**

**FDP_ACF.1.2** **The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed [assignment: *rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*].**

Dependencies : **FDP_ACC.1 Subset Object Access Control**

### FDP_ACF.2 Multiple Security Attribute Access Control

Hierarchical to: FDP_ACF.1

**FDP_ACF.2.1** The TSF shall enforce the [assignment: *access control SFP*] to objects based on [assignment: ***multiple*** *attributes*, ***multiple*** *named groups of attributes*].

**FDP_ACF.2.2** The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed [assignment: *rules governing access among controlled subjects and controlled objects using controlled operations on controlled objects*].

Dependencies : FDP_ACC.1 Subset Object Access Control

### FDP_ACF.3 Access Authorisation

Hierarchical to: no other components.

**FDP_ACF.3.1** **The TSF shall enforce the [assignment: *access control SFP*] to provide the ability to explicitly grant access based on the value of security attributes of subjects and objects.**

Dependencies : **FDP_ACC.1 Subset Object Access Control**

### FDP_ACF.4 Access Authorisation and Denial

Hierarchical to: FDP_ACF.3

**FDP_ACF.4.1** The TSF shall enforce the [assignment: *access control SFP*] to provide the ability to explicitly grant access based on the value of security attributes of subjects and objects.

**FDP_ACF.4.2** **The TSF shall enforce the [assignment: *access control SFP*] to provide the ability to explicitly deny access based on the value of security attributes of subjects and objects.**

Dependencies : FDP_ACC.1 Subset Object Access Control

### FDP_ACF.5 Fixed Access Control

Hierarchical to: no other components.

**FDP_ACF.5.1** **The TSF shall enforce the [assignment: *access control SFP*] so that the security attributes of the controlled objects and subjects cannot be changed.**

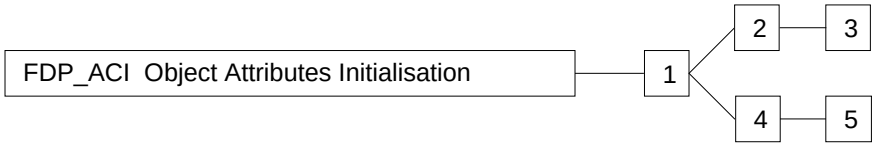
Dependencies : **FDP_ACC.1 Subset Object Access Control**

# FDP_ACI Object Attributes Initialisation

## Family behaviour

177 This family defines the rules for the initial values of relevant access control security attributes for objects as required for each access control SFP enforced by the TOE. These rules address the need for objects to be protected appropriately by default.

## Component levelling




178 FDP_ACI.1 Static Attribute Initialisation, requires that the TSF provide default values for relevant security attributes, but that no mechanisms need to be provided to modify these defaults.

179 FDP_ACI.2 Administrator Defined Attribute Initialisation, requires that the default values be modifiable by the authorised administrator.

180 FDP_ACI.3 User Defined Attribute Initialisation, requires that the default values be modifiable by authorised users.

181 FDP_ACI.4 Safe Access Control Attribute Initialisation, requires that the TSF enforce rules on the acceptable settings of default values.

182 FDP_ACI.5 Safe Access Control Attribute Modification, requires that the TSF enforce rules on the acceptable modifications of default values.

## Audit :

183 The following actions shall be audited if FAU_GEN Security Audit Data Generation is included in the PP/ST:

a) Minimal: Successful changes to default object attributes.

b) Minimal: Successful overriding of the default object attributes.

c) Basic: Any changes or overriding of the default object attributes including identification of which default object attributes have been changed or overridden.

d) Detailed: Capture of the actual values of each security attribute changed or overridden.

### FDP_ACI.1 Static Attribute Initialisation

Hierarchical to: no other components.

**FDP_ACI.1.1** **The TSF shall enforce the [selection: *access control SFP, information flow control SFP*] to provide [selection: *restrictive, permissive, other property*] default values for object security attributes that are used to enforce the *SFP*.**

**FDP_ACI.1.2** **The TSF shall allow the specification of alternate initial values to override the default values when an object is created.**

Dependencies : **[FDP_ACC.1 Subset Object Access Control**, or
**FDP_IFC.1 Subset Information Flow Control]**

### FDP_ACI.2 Administrator Defined Attribute Initialisation

Hierarchical to: FDP_ACI.1

**FDP_ACI.2.1** The TSF shall enforce the [selection: *access control SFP, information flow control SFP*] to provide [selection: *restrictive, permissive, other property*] default values for object security attributes that are used to enforce the *SFP*.

**FDP_ACI.2.2** The TSF shall allow the specification of alternate initial values to override the default values when an object is created.

**FDP_ACI.2.3** **The TSF shall restrict modification of these default values to the authorised administrator.**

Dependencies : [FDP_ACC.1 Subset Object Access Control, or
FDP_IFC.1 Subset Information Flow Control]
**FPT_TSA.1 Basic Security Administration**
**FPT_TSU.1 Enforcement of Administrative Guidance**

### FDP_ACI.3 User Defined Attribute Initialisation

Hierarchical to: FDP_ACI.2

**FDP_ACI.3.1** The TSF shall enforce the [selection: *access control SFP, information flow control SFP*] to provide [selection: *restrictive, permissive, other property*] default values for object security attributes that are used to enforce the *SFP*.

**FDP_ACI.3.2** The TSF shall allow the specification of alternate initial values to override the default values when an object is created.

**FDP_ACI.3.3** The TSF shall **allow** modification of these default values to the authorised administrator.

**FDP_ACI.3.4** The TSF shall **provide authorised users the capability to modify the default values of their related attributes**.

Dependencies : [FDP_ACC.1 Subset Object Access Control or
FDP_IFC.1 Subset Information Flow Control]
FPT_TSA.1 Basic Security Administration
FPT_TSU.1 Enforcement of Administrative Guidance

### FDP_ACI.4 Safe Access Control Attribute Initialisation

Hierarchical to: FDP_ACI.1

**FDP_ACI.4.1** The TSF shall enforce the [selection: *access control SFP, information flow control SFP*] to provide [selection: *restrictive, permissive, other property*] default values for object security attributes that are used to enforce the *SFP*.

**FDP_ACI.4.2** The TSF shall allow the specification of alternate initial values to override the default values when an object is created.

**FDP_ACI.4.3** **The TSF shall enforce the [selection: *access control SFP, information flow control SFP*] to accept only valid initial object security attribute values.**

Dependencies : **[**FDP_ACF Access Control Functions, or
FDP_IFC Information Flow Control Policy**]**
**FPT_TSU.1 Enforcement of Administrative Guidance**

### FDP_ACI.5 Safe Access Control Attribute Modification

Hierarchical to: FDP_ACI.4

**FDP_ACI.5.1** The TSF shall enforce the [selection: *access control SFP, information flow control SFP*] to provide [selection: *restrictive, permissive, other property*] default values for object security attributes that are used to enforce the *SFP*.

**FDP_ACI.5.2** The TSF shall allow the specification of alternate initial values to override the default values when an object is created.

**FDP_ACI.5.3** The TSF shall enforce the [selection: *access control SFP, information flow control SFP*] to accept only valid initial **or modified** object security attribute values.

Dependencies : **[**FDP_ACF Access Control Functions, or
FDP_IFC Information Flow Control Policy**]**
FPT_TSU.1 Enforcement of Administrative Guidance

# FDP_ETC Export to Outside TSF Control

Family behaviour

184 This family defines functions for exporting user data from the TOE such that its security attributes and protection can be preserved. It is concerned with limitations on export, the form of the information (e.g., machine-readable, human-readable), user specification of security attributes, and association of security attributes with the exported information.

Component levelling


185 FDP_ETC.1 Export of User Data Without Security Attributes requires that security attributes correctly represent the object and are bound to objects exported from the TSF.

186 FDP_ETC.2 Export of User Data With Security Attributes requires that security attributes are somehow communicated to the authorised user at the destination to accurately represent the object exported from the TSF.

Audit :

187 The following events shall be auditable if FAU_GEN Security Audit is included in the PP/ST:

a) Minimal: Successful export of information.

b) Basic: All attempts to export information.

## FDP_ETC.1 Export of User Data Without Security Attributes

Hierarchical to: no other components.

**FDP_ETC.1.1 The TSF shall enforce the [selection: *access control SFP, information flow control SFP*] for information exported outside the TSC via a function that does not provide the information's corresponding security attributes.**

Dependencies : **[FDP_ACC.1 Subset Object Access Control, or**
**FDP_IFC.1 Subset Information Flow Control]**

## FDP_ETC.2 Export of User Data With Security Attributes

Hierarchical to: FDP_ETC.1

**FDP_ETC.2.1** The TSF shall enforce the [selection: *access control SFP, information flow control SFP*] for information exported outside the TSC via a function **that provides** the information's corresponding security attributes.

**FDP_ETC.2.2** **The TSF shall enforce that the security attributes used for the purpose of access control, when exported outside the TSC, accurately and unambiguously represent the corresponding security attributes.**

**FDP_ETC.2.3** **The TSF shall ensure that the security attributes used for the purpose of access control, when exported outside the TSC, are unambiguously associated with the information exported.**

**FDP_ETC.2.4** **The TSF shall enforce [assignment: *additional exportation control rules*] when information is exported from the TSC.**

Dependencies : [FDP_ACC.1 Subset Object Access Control, or
FDP_IFC.1 Subset Information Flow Control]
**[FTP_ITC.1 Inter-TSF Trusted Channel or**
**FTP_TRP.1 Trusted Path]**
**FPT_TDC.1 Inter-TSF Basic TSF Data Consistency**

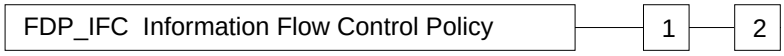
# FDP_IFC Information Flow Control Policy

## Family behaviour

188 This family defines the scope of control of the information flow control policies that enforce rules preventing the unauthorised flow of information among subjects and objects. This family is distinct from FDP_IFF Information Flow Control Functions in order to separate policy from mechanism. The family defines a set of named information flow control SFPs; and, for each, specifies the scope of control.

189 The TSF mechanism controls the flow in accordance with the information flow control SFP regardless of the operations invoked. Operations which would change the information flow are not permitted as this would be in violation of the SFP.

## Component levelling




190 FDP_IFC.1 Subset Information Flow Control allows the specification of a TSP, the information flow portion of which is made up of multiple information flow control SFPs where at least one information flow control SFP does not cover all operations, objects, or subjects managed by the TSF.

191 FDP_IFC.2 Complete Information Flow Control allows the specification of a TSP, the information flow portion of which is made up of multiple information flow control SFPs where every information flow control SFP covers all operations on subjects and objects managed by the information flow control SFP. Further, the combination of the information flow control SFPs is such that every subject and object is covered by at least one information flow control SFP. This element ensures that some form of information flow control is in place for all operations. In conjunction with the FPT_RVM.1 component, this gives the “always invoked” aspect of a reference monitor.

## FDP_IFC.1 Subset Information Flow Control

Hierarchical to: no other components.

**FDP_IFC.1.1** **The TSF shall enforce the [assignment: *information flow control SFP*] on [assignment: *list of subjects, objects and operations among subjects and objects covered by the SFP*].**

Dependencies : **FDP_IFF.1 Simple Security Attributes**

## FDP_IFC.2 Complete Information Flow Control

Hierarchical to: FDP_IFC.1

**FDP_IFC.2.1** The TSF shall enforce the [assignment: *information flow control SFP*] on [assignment: *list of subjects* and *objects*] and **all** operations among subjects and objects covered by the SFP.

**FDP_IFC.2.2** **The TSF shall ensure that all objects and operations within the TSC are covered by at least one information flow control SFP.**

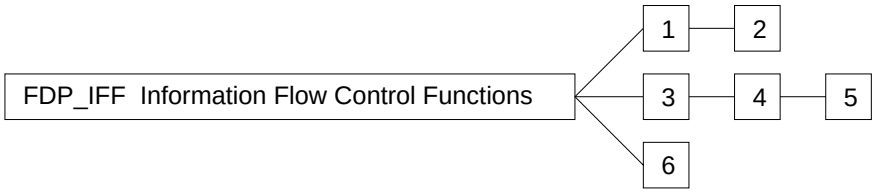
Dependencies : FDP_IFF.1 Simple Security Attributes

# FDP_IFF Information Flow Control Functions

## Family behaviour

192 This component specifies the requirements on functions with respect to the information flow control SFPs. It consists of two "trees": one addressing the common information flow function issues, and a second addressing illicit information flow channels (i.e., covert channels) with respect to one or more information flow control SFPs. This division arises because the issues concerning illicit information flows are, in some sense, orthogonal to the rest of an information flow control SFP. Illicit information flows are flows in violation of policy; thus they are not a policy issue (if they were explicitly allowed by the policy, they would not be illicit).

## Component levelling




193 FDP_IFF.1 Simple Security Attributes requires security attributes on containers of information, and on active recipients of information. It specifies the key rules that must be enforced by the function, and describes how security attributes are derived by the function.

194 FDP_IFF.2 Hierarchical Security Attributes expand on the requirements of FDP_IFF.1 Simple Security Attributes by requiring that all information flow control SFPs in the TSP use hierarchical security attributes that form a lattice.

195 FDP_IFF.3 Limited Illicit Information Flows requires the SFP cover illicit information flows, but not necessarily eliminate them.

196 FDP_IFF.4 Partial Elimination of Illicit Information Flows requires the SFP to cover the elimination of some (but not necessarily all) illicit information flows.

197 FDP_IFF.5 No Illicit Information Flows requires SFP cover the elimination of all illicit information flows.

198 FDP_IFF.6 Illicit Information Flow Monitoring requires the SFP to monitor illicit information flows for specified and maximum capacities.

## Audit :

199 The following actions shall be auditable if FAU_GEN Security Audit Data Generation is included in a PP/ST:

a) Minimal: Decisions to permit requested information flows.

b) Basic: All decisions on requests for information flow.

c) Basic: The use of identified illicit information flow channels.

d) Detailed: The specific security attributes used in making an information flow enforcement decision.

e) Detailed: Some specific subsets of the information which has flowed based upon policy goals (e.g. auditing of downgraded material).

f) Detailed: The use of identified illicit information flow channels with estimated maximum capacity exceeding a specified value.

### FDP_IFF.1 Simple Security Attributes

Hierarchical to: no other components.

**FDP_IFF.1.1** **The TSF shall enforce the [assignment: *information flow control SFP*] to enforce at least the following types of subject and object security attributes [assignment: *specification of the minimum number and type of security attributes*].**

**FDP_IFF.1.2** **The TSF shall enforce an information flow between a subject and a controlled object via a controlled operation if the following rules hold [assignment: *by operation, the security attribute-based relationship that must hold between subject and object security attributes*].**

**FDP_IFF.1.3** **The TSF shall enforce the [assignment: *additional information flow control SFP rules*].**

**FDP_IFF.1.4** **The TSF shall enforce the following [assignment: *list of additional SFP capabilities*].**

Dependencies : **FDP_IFC.1 Subset Information Flow Control**

### FDP_IFF.2 Hierarchical Security Attributes

Hierarchical to: FDP_IFF.1

**FDP_IFF.2.1** The TSF shall enforce the [assignment: *information flow control SFP*] to enforce at least the following types of subject and object security attributes [assignment: *specification of the minimum number and type of security attributes*].

**FDP_IFF.2.2** The TSF shall permit an information flow between a subject and a controlled object via a controlled operation if the following rules, **based on the ordering relationships between security attributes** hold; [assignment: *by operation, the security attribute based relationship that must hold between subject and object security attributes*].

**FDP_IFF.2.3** The TSF shall enforce the [assignment: *additional information flow control SFP rules*].

**FDP_IFF.2.4** The TSF shall enforce the following [assignment: *list of additional SFP capabilities*]

**FDP_IFF.2.5** **The TSF shall enforce the following relationships for any two valid security attributes:**

a) **There exists an ordering function that, given two valid security attributes, determines if the security attributes are equal, if one security attribute is greater than the other, or if the security attributes are incomparable; and**

b) **There exists a "least upper bound" in the set of security attributes, such that, given any two valid security attributes, there is a valid security attribute that is greater than or equal to the two valid security attributes; and**

c) **There exists a "greatest lower bound" in the set of security attributes, such that, given any two valid security attributes, there is a valid security attribute that is not greater than the two valid security attributes.**

Dependencies : FDP_IFC.1 Subset Information Flow Control

### FDP_IFF.3 Limited Illicit Information Flows

Hierarchical to: no other components.

**FDP_IFF.3.1** **The TSF shall enforce the [assignment: *information flow control SFP*] to limit the capacity of [assignment: *types of illicit information flows*] to a [assignment: *maximum capacity*].**

Dependencies : **AVA_CCA.1 Covert channel analysis**
**FDP_IFC.1 Subset Information Flow Control**

### FDP_IFF.4 Partial Elimination of Illicit Information Flows

Hierarchical to: FDP_IFF.3

**FDP_IFF.4.1** The TSF shall enforce the [assignment: *information flow control SFP*] to limit the capacity of [assignment: *non-empty list of types of illicit information flows*] to a [assignment: *maximum capacity*].

**FDP_IFF.4.2** **The TSF shall prevent the following types of [assignment: *non-empty list of types of illicit information flows*].**

Dependencies : AVA_CCA.1 Covert channel analysis, or
FDP_IFC.1 Subset Information Flow Control

### FDP_IFF.5 No Illicit Information Flows

Hierarchical to: FDP_IFF.4

**FDP_IFF.5.1** **The TSF shall ensure that no illicit information flows exist to circumvent [assignment: *name of information flow control SFP*].**

Dependencies : AVA_CCA.1 Covert channel analysis, or
FDP_IFC.1 Subset Information Flow Control

### FDP_IFF.6 Illicit Information Flow Monitoring

Hierarchical to: no other components.

**FDP_IFF.6.1** **The TSF shall enforce the [assignment: *information flow control SFP*] to monitor the [assignment: *list of types of illicit information flows*] for the** [**assignment: *specified capacity*].**

**FDP_IFF.6.2** **The TSF shall monitor the [assignment: *list of types of illicit information flows*] for the** [**assignment: *maximum capacity*].**

Dependencies : **AVA_CCA.1 Covert channel analysis**, or
**FDP_IFC.1 Subset Information Flow Control**

# FDP_ITC Import from Outside TSF Control

## Family behaviour

200 This family defines the mechanisms for introduction of information into the TOE such that it has appropriate security attributes and is appropriately protected. It is concerned with limitations on importation, the form of the information (e.g., human-entered, machine-readable), determination of desired security attributes, and interpretation of security attributes associated with the information.

## Component levelling

| | | |
|---|---|---|
| FDP_ITC Import from Outside TSF Control | 1 | 2 |

201 This family contains two components to address the preservation of security attributes of imported user data for access control and information control policies.

202 Component FDP_ITC.2 Import of User Data with Security Attributes requires that security attributes correctly represent the object and are bound to the objects imported from outside the TSC.

203 Component FDP_ITC.1 Import of User Data Without Security Attributes requires that the security attributes correctly represent the information and are communicated to the object to the destination separately from the object.

## Audit :

204 The following events shall be auditable if FAU_GEN Security Audit is included in the PP/ST:

a) Minimal: Successful import of information, including any security attributes received.

b) Minimal: All attempts to import information, including any security attributes provided.

c) Detailed: The specification of security attributes for imported information supplied by an authorised user.

## FDP_ITC.1 Import of User Data Without Security Attributes

Hierarchical to: no other components.

**FDP_ITC.1.1 The TSF shall enforce the [selection: *access control SFP, information flow control SFP*] for information imported from outside the TSC by the TSF via a function that does not provide reliable security attributes.**

**FDP_ITC.1.2** **The TSF shall allow an authorised user to supply the security attributes for the information received.**

**FDP_ITC.1.3** **The TSF shall provide the following [assignment: *additional importation control rules*] when information controlled under the SFP is imported from outside the TSC.**

Dependencies : [**FDP_ACC.1 Subset Object Access Control**, or
**FDP_IFC.1 Subset Information Flow Control**]

## FDP_ITC.2 Import of User Data with Security Attributes

Hierarchical to: FDP_ITC.1

**FDP_ITC.2.1** The TSF shall enforce the [selection: *access control SFP, information flow control SFP*] for information imported from outside the TSC by the TSF via a mechanism **that provides** reliable security attributes.

**FDP_ITC.2.2** **The TSF shall ensure that the protocol used provides for the unambiguous association between the security attributes and the information received.**

**FDP_ITC.2.3** **The TSF shall ensure that interpretation of the security attributes of the imported information is as intended by the source TSF.**

**FDP_ITC.2.4** The TSF shall enforce the following [assignment: *additional importation control rules*] when information controlled under the SFP is imported from outside the TSC.

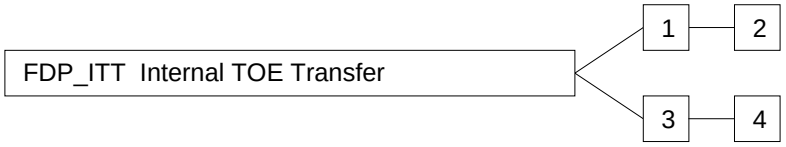
Dependencies : [FDP_ACC.1 Subset Object Access Control, or
FDP_IFC.1 Subset Information Flow Control]
**[FTP_ITC.1 Inter-TSF Trusted Channel or**
**FTP_TRP.1 Trusted Path]**
**FPT_TDC.1 Inter-TSF Basic TSF Data Consistency**

## FDP_ITT Internal TOE Transfer

### Family behaviour

205 This family provides requirements that address protection of user data when it is transferred between parts of a TOE across an internal channel. This may be contrasted with the FDP_UCT and FDP_UIT families, which provide protection for user data when it is transferred between distinct TSFs across an external channel, and FDP_ACI and FDP_ITC, which address transfer of data to or from a non-TSF controlled environment.

### Component levelling




206 FDP_ITT.1 Basic Internal Transfer Protection requires that user data be protected when transmitted between parts of the TOE.

207 FDP_ITT.2 Transmission Separation by Attribute requires separation of data based on SFP-relevant attributes in addition to the first component.

208 FDP_ITT.3 Integrity Monitoring is distinct from the first two components, and requires that The SF monitor user data transmitted between parts of the TOE for identified integrity errors.

209 FDP_ITT.4 Attribute-Based Integrity Monitoring expands on the third component by allowing the form of integrity monitoring to differ by SFP-relevant attribute.

### Audit :

210 The following events should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST:

a) Minimal: Successful transfers of user data, including identification of the transmission channel and the integrity protection method used.

b) Basic: All attempts to transfer user data, including identification of the transmission channel used, the integrity protection method used, and the error which occurred.

c) Basic: Unauthorised attempts to change the integrity protection method.

d) Basic: Unauthorised attempts to configure the separation mechanism.

e) Detailed: The action taken upon detection of an integrity error.

### FDP_ITT.1 Basic Internal Transfer Protection

Hierarchical to: no other components.

**FDP_ITT.1.1** **The TSF shall enforce the [selection: *access control SFP, information flow control SFP*] to protect user data from [selection: *disclosure, modification, non-availability*] when it is transmitted between physically-separated parts of the TOE.**

**FDP_ITT.1.2** **If the TSF provides multiple methods to protect user data during transmission between physically-separated parts of the TOE, the TSF shall provide authorised administrators with the ability to select the method used.**

**FDP_ITT.1.3** **The TSF shall restrict the ability to configure the separation mechanism to authorised administrators.**

Dependencies : **[FDP_ACC.1 Subset Object Access Control, or**
**FDP_IFC.1 Subset Information Flow Control]**
**FPT_TSA.1 Basic Security Administration**

### FDP_ITT.2 Transmission Separation by Attribute

Hierarchical to: FDP_ITT.1

**FDP_ITT.2.1** The TSF shall enforce the [selection: *access control SFP, information flow control SFP*] to protect user data from [selection: disclosure, modification, non-availability] when it is transmitted between physically-separated parts of the TOE.

**FDP_ITT.2.2** **The TSF shall provide separate transmission channels for data controlled by the SFP and transmitted between physically-separated parts of the TOE based on the following [assignment: *security attributes that require separate transmission channels*].**

**FDP_ITT.2.3** If the TSF provides multiple approved methods to protect user data during transmission between physically-separated parts of the TOE, the TSF shall provide authorised administrators with the ability to select the method used.

**FDP_ITT.2.4** The TSF shall restrict the ability to configure the separation mechanism to authorised administrators.

Dependencies : [FDP_ACC.1 Subset Object Access Control, or
FDP_IFC.1 Subset Information Flow Control]
FPT_TSA.1 Basic Security Administration

### FDP_ITT.3 Integrity Monitoring

Hierarchical to: no other components.

**FDP_ITT.3.1** **The TSF shall enforce the [selection: *access control SFP, information flow control SFP*] to monitor user data transmitted between physically-separated parts of the TOE for [assignment: *integrity errors*].**

**FDP_ITT.3.2** **Upon detection of a data integrity error, the TSF shall [assignment: *specify the action to be taken upon integrity error*].**

Dependencies : **[FDP_ACC.1 Subset Object Access Control, or**
**FDP_IFC.1 Subset Information Flow Control]**
**FDP_ITT.1 Basic Internal Transfer Protection**

### FDP_ITT.4 Attribute-Based Integrity Monitoring

Hierarchical to: FDP_ITT.3

**FDP_ITT.4.1** The TSF shall enforce the [selection: *access control SFP, information flow control SFP*] to monitor user data **controlled by the SFP and** transmitted between physically-separated parts of the TOE for [assignment: *integrity errors*], **based on the following [assignment: *security attributes that require separate transmission channels*].**

**FDP_ITT.4.2** Upon detection of a data integrity error, the TSF shall [assignment: *specify the action to be taken upon integrity error*].

Dependencies : [FDP_ACC.1 Subset Object Access Control, or
FDP_IFC.1 Subset Information Flow Control]
**FDP_ITT.2 Transmission Separation by Attribute**

# FDP_RIP Residual Information Protection

Family behaviour

211 This family addresses the need to ensure that deleted information is no longer accessible, and that newly created objects do not contain information that should not be accessible. This family requires protection for information that has been logically deleted or released, but may still be physically present within the TOE.

Component levelling


212 FDP_RIP.1 Subset Residual Information Protection on Allocation requires that the TSF ensure that any residual information content of any resources being allocated to a defined subset of the objects in the TSC is unavailable.

213 FDP_RIP.2 Subset Residual Information Protection on Deallocation requires that the TSF make unavailable any residual information content of a resource when the resource is deallocated from a defined subset of the objects in the TSC.

214 FDP_RIP.3 Full Residual Information Protection on Allocation requires that the TSF ensure that any residual information content of any resources being allocated to an object is unavailable.

215 FDP_RIP.4 Full Residual Information Protection on Deallocation requires that the TSF make unavailable any residual information content of a resource when the resource is deallocated from an object in the TSC.

## FDP_RIP.1 Subset Residual Information Protection on Allocation

Hierarchical to: no other components.

**FDP_RIP.1.1** **The TSF shall ensure that upon the allocation of a resource to [assignment: *list of objects*] any previous information content is unavailable.**

Dependencies : No dependencies.

## FDP_RIP.2 Subset Residual Information Protection on Deallocation

Hierarchical to: **FDP_RIP.1**

**FDP_RIP.2.1** The TSF shall ensure that upon the **deallocation** of a resource **from** [assignment: *list of objects*] any previous information content is **made** unavailable.

Dependencies : No dependencies.

### FDP_RIP.3 Full Residual Information Protection on Allocation

Hierarchical to: FDP_RIP.1

**FDP_RIP.3.1** The TSF shall ensure that upon the allocation of a resource to **all objects** any previous information content is unavailable.

Dependencies : No dependencies.

### FDP_RIP.4 Full Residual Information Protection on Deallocation

Hierarchical to: **FDP_RIP.2 and FDP_RIP.3**

**FDP_RIP.4.1** The TSF shall ensure that upon the deallocation of a resource from **all objects** any previous information content is made unavailable.

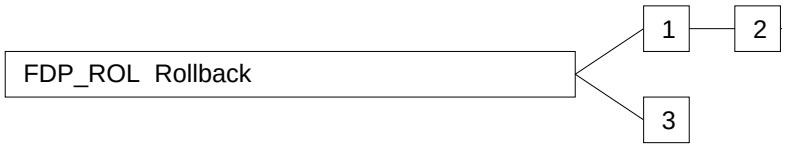
Dependencies : No dependencies.

# FDP_ROL Rollback

Family behaviour

216 The rollback operation involves undoing the last operation or a series of operations, bounded by some limit, such as a period of time, and return to a previous known state. Rollback provides the ability to undo the effects of an operation or series of operations to preserve the integrity of the user data.

Component levelling




217 FDP_ROL.1 Basic Rollback addresses a need to roll back or undo a limited number of operations within the defined bounds.

218 FDP_ROL.2 Advanced Rollback addresses the need to roll back or undo all operations within the defined bounds.

219 FDP_ROL.3 Administrative Rollback addresses the need for administrators change the rollback boundaries.

Audit :

220 The following should be audited if FAU_GEN Security Audit Data Generation is specified in the PP/ST:

a) Minimal: All successful rollback operations.

b) Basic: All attempts to perform rollback operations.

c) Detailed: All attempts to perform rollback operations, including identification of the types of operations rolled back.

## FDP_ROL.1 Basic Rollback

Hierarchical to: no other components.

**FDP_ROL.1.1** **The TSF shall enforce [selection: *access control SFP, information flow control SFP*] to permit the [assignment: *list of authorised users* and *subjects*] to rollback the [assignment: *list of operations*] on the [assignment: *list of objects*].**

**FDP_ROL.1.2** **The TSF shall permit operations to be rolled back within the [assignment: *boundary limit to which rollback may be performed*].**

Dependencies : **FIA_UID.1 Basic User Identification**

### FDP_ROL.2 Advanced Rollback

Hierarchical to: FDP_ROL.1

**FDP_ROL.2.1** The TSF shall enforce [selection: *access control SFP*, *information flow control SFP*] to permit the [assignment: *list of authorised users* and *subjects*] to rollback **all the operations** on the [assignment: *list of objects*].

**FDP_ROL.2.2** The TSF shall permit operations to be rolled back within the [assignment: *boundary limit to which rollback may be performed*].

Dependencies : FIA_UID.1 Basic User Identification

### FDP_ROL.3 Administrative Rollback

Hierarchical to: no other components.

**FDP_ROL.3.1** **The TSF shall enforce [selection: *access control SFP, information flow control SFP*] to permit an authorised administrator to change the [assignment: *boundary in which rollback may be performed*].**

Dependencies : **FPT_TSA.1 Basic Security Administration**

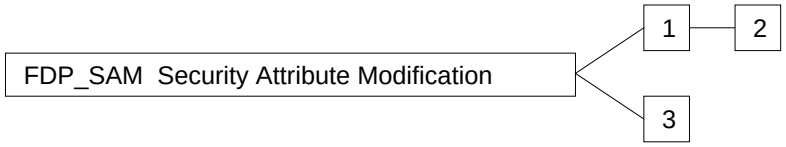
**FDP_ROL.1 Basic Rollback**

# FDP_SAM Security Attribute Modification

Family behaviour

221 This family defines the rules for modifying values of security attributes relevant to a user data protection policy.

Component levelling




222 FDP_SAM.1 Administrator Attribute Modification, limits the modification of relevant security attributes associated with objects, users, or subjects to only authorised administrators.

223 FDP_SAM.2 User Attribute Modification, increases the functionality available by allowing the modification of relevant security attributes by any authorised user.

224 FDP_SAM.3 Safe Attribute Modification, ensures that the values provided are valid for the specified security attributes.

Audit :

225 The following events should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST:

a) Minimal: Successful modification of security attributes, including the identity of the target of the modification.

b) Basic: All attempts to modify security attributes, including the identity of the target of the modification attempt.

c) Detailed: All attempts to modify security attributes, including the identity of the target of the modification attempt and the new values of modified security attributes.

## FDP_SAM.1 Administrator Attribute Modification

Hierarchical to: no other components.

FDP_SAM.1.1 **The TSF shall enforce [selection: *access control SFP, information flow control SFP*] to provide authorised administrators with the ability to modify [assignment: *list of security attributes*].**

Dependencies : **FPT_TSA.1 Basic Security Administration**
**[FDP_ACC.1 Subset Object Access Control, or**
**FDP_IFC.1 Subset Information Flow Control]**

### FDP_SAM.2 User Attribute Modification

Hierarchical to: FDP_SAM.1

**FDP_SAM.2.1** The TSF shall enforce [selection: *access control SFP, information flow control SFP*] to provide authorised **users** with the ability to modify [assignment: *list of security attributes*].

Dependencies : FPT_TSA.1 Basic Security Administration
[FDP_ACC.1 Subset Object Access Control, or
FDP_IFC.1 Subset Information Flow Control]

### FDP_SAM.3 Safe Attribute Modification

Hierarchical to: no other components.

**FDP_SAM.3.1** **The TSF shall enforce [selection: *access control SFP, information flow control SFP*] to verify that the modified values are valid when changes are made to the [assignment: *list of security attributes*].**

Dependencies : **FDP_SAM.1 Administrator Attribute Modification**
**[FDP_ACC.1 Subset Object Access Control, or**
**FDP_IFC.1 Subset Information Flow Control]**

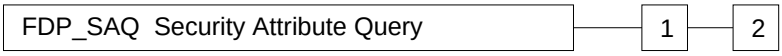
# FDP_SAQ Security Attribute Query

Family behaviour

226 This family addresses the need for authorised users, and subjects operating on behalf of authorised users, to query security attributes.

227 This family specifies two hierarchical components concerning operations for viewing relevant data protection security attributes of objects, subjects, and users. For each, the levelling is based on the scope of the required functionality:

Component levelling




228 FDP_SAQ.1 Administrator Attribute Query limits the query of relevant security attributes associated with objects, users or subjects to authorised administrators.

229 FDP_SAQ.2 User Attribute Query, increases the functionality available by allowing the querying of relevant security attributes by any authorised user.

Audit :

230 The following events should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST:

a) Minimal: Successful query of security attributes, including the identity of the target of the query.

b) Basic: All attempts to query security attributes, including the identity of the target of the query attempt.

## FDP_SAQ.1 Administrator Attribute Query

Hierarchical to: no other components.

**FDP_SAQ.1.1** **The TSF shall enforce the [selection: *access control SFP, information flow control SFP*] to provide the authorised administrator with the ability to query [assignment: *list of security attributes*] values.**

Dependencies : **FPT_TSA.1 Basic Security Administration**
**[FDP_ACC.1 Subset Object Access Control or**
**FDP_IFC.1 Subset Information Flow Control]**

### FDP_SAQ.2 User Attribute Query

Hierarchical to: FDP_SAQ.1.

**FDP_SAQ.2.1** The TSF shall enforce the [selection: *access control SFP, information flow control SFP*] to provide the authorised **users** with the ability to query [assignment: *list of security attributes*] values.

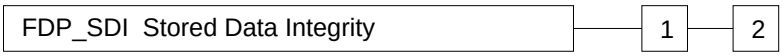
Dependencies : [FDP_ACC.1 Subset Object Access Control or
FDP_IFF.1 Simple Security Attributes]

# FDP_SDI Stored Data Integrity

Family behaviour

231 This family provides requirements that address protection of user data while it is stored within the TSC. Integrity errors may affect user data stored in memory, or in a storage device. This family differs from FDP_ITT Internal TOE Transfer which protects the user data from integrity errors while being transferred within the TOE.

Component levelling




232 FDP_SDI.1 Stored Data Integrity Monitoring requires that the SF monitor user data stored within the TSC for identified integrity errors.

233 FDP_SDI.2 Stored Data Attribute-Based Integrity Monitoring adds the additional capability to the first component by allowing the form of integrity monitoring to differ by user data attribute.

Audit :

234 The following events should be auditable if FAU_GEN Security Audit is included in the PP/ST:

a) Minimal: Successful attempts to check the integrity of user data, including an indication of the results of the check.

b) Basic: All attempts to check the integrity of user data, including an indication of the results of the check, if performed.

c) Basic: Unauthorised attempts to change the integrity protection method.

d) Basic: Unauthorised attempts to configure the separation mechanism.

e) Detailed: The type of integrity error which occurred.

f) Detailed: The action taken upon detection of an integrity error.

## FDP_SDI.1 Stored Data Integrity Monitoring

Hierarchical to: no other components.

**FDP_SDI.1.1 The TSF shall ensure that upon detection of a data integrity error of [assignment: *list of objects*], the TSF shall [assignment: *action to be taken*].**

Dependencies : No dependencies.

## FDP_SDI.2 Stored Data Attribute-Based Integrity Monitoring

Hierarchical to: FDP_SDI.1

**FDP_SDI.2.1** **The TSF shall monitor user data stored within the TSC for [assignment: *integrity errors*] on all objects, based on the following [assignment: *user data attributes*].**

**FDP_SDI.2.2** **Upon detection of a data integrity error, the TSF shall [assignment: specify the action to be taken upon integrity error].**

Dependencies : No dependencies.

# FDP_UCT Inter-TSF User Data Confidentiality Transfer Protection

Family behaviour

235 This family defines the requirements for ensuring the confidentiality of user data when it is transferred using an external channel between distinct TOEs or users on distinct TOEs.

Component levelling


236 In FDP_UCT.1 Basic Data Exchange Confidentiality, the goal is to provide protection from disclosure of user data while in transit.

Audit :

237 The following events should be audited, if FAU_GEN Security Audit Data Generation is included in the PP/ST.

a) Minimal: The identity of any user or subject using the data exchange mechanisms.

b) Basic: The identity of any unauthorised user or subject attempting to use the data exchange mechanisms.

c) Basic: A reference to the names or other indexing information useful in identifying that data that was transmitted or received. This could include security attributes associated with the information.

d) Detailed: Partial or complete content of data transmitted.

## FDP_UCT.1 Basic Data Exchange Confidentiality

Hierarchical to: no other components.

**FDP_UCT.1.1 The TSF shall enforce the [selection: *access control SFP*] to be able to [selection*: transmit, receive*] objects in a manner protected from unauthorised disclosure.**

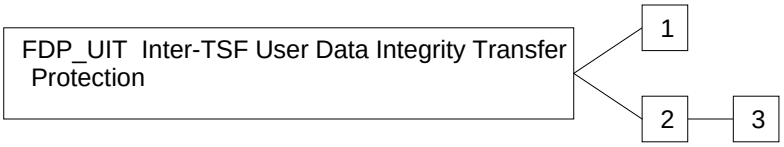
Dependencies : **[FTP_ITC.1 Inter-TSF Trusted Channel, or**
**FTP_TRP.1 Trusted Path]**
**FDP_ACC.1 Subset Object Access Control**

# FDP_UIT Inter-TSF User Data Integrity Transfer Protection

## Family behaviour

238 This family defines the requirements for protecting user data in transit between TSFs and recovering from detectable errors.

## Component levelling




239 FDP_UIT.1 Data Exchange Integrity addresses detection of modifications, deletions, insertions, and replay errors of the user data transmitted.

240 FDP_UIT.2 Destination Data Exchange Recovery addresses recovery of the original user data by the receiving TSF on its own without any help from the source TSF.

241 FDP_UIT.3 Source Data Exchange Recovery addresses recovery of the original user data by the receiving TSF with help from the source TSF.

## Audit :

242 The following events shall be audited, if FAU_GEN Security Audit is included in the PP/ST.

a) Minimal: The identity of any user or subject using the data exchange mechanisms.

b) Minimal: Whether any modifications to transmitted data were detected, if they were not also corrected.

c) Basic: The identity of any user or subject attempting to use the data exchange mechanisms, but who is unauthorised to do so.

d) Basic: A reference to the names or other indexing information useful in identifying that data that was transmitted or received. This could include security attributes associated with the information.

e) Basic: Whether any modifications to data were corrected.

f) Basic: Any identified attempts to block transmission of user data.

g) Detailed: The types and or effects of any detected modifications of transmitted data.

h) Detailed: Partial or complete content of data transmitted.

### FDP_UIT.1 Data Exchange Integrity

Hierarchical to: no other components.

**FDP_UIT.1.1** **The TSF shall enforce the [selection: *access control SFP, information flow control SFP*] to be able to [selection*: transmit, receive*] objects in a manner protected from undetectable [selection: *modification, deletion, insertion, replay*] errors.**

**FDP_UIT.1.2** **The TSF shall be able to determine on receipt of an object, whether [selection: *modification, deletion, insertion, or replay*] has occurred.**

Dependencies : **[FTP_ITC.1 Inter-TSF Trusted Channel, or**
**FTP_TRP.1 Trusted Path]**
**[FDP_ACC.1 Subset Object Access Control, or**
**FDP_IFC.1 Subset Information Flow Control]**

### FDP_UIT.2 Destination Data Exchange Recovery

Hierarchical to: no other components.

**FDP_UIT.2.1** **The TSF shall enforce the [selection: *access control SFP, information flow control SFP*] to be able to recover from [assignment: *list of recoverable errors*].**

Dependencies : **[FDP_ACC.1 Subset Object Access Control, or**
**FDP_IFC.1 Subset Information Flow Control]**
**FTP_ITC.1 Inter-TSF Trusted Channel**

### FDP_UIT.3 Source Data Exchange Recovery

Hierarchical to: FDP_UIT.2

**FDP_UIT.3.1** The TSF shall enforce the [selection: *access control SFP, information flow control SFP*] to be able to recover from [assignment: *list of recoverable errors*] **with the help of the source TSF**.

Dependencies : [FDP_ACC.1 Subset Object Access Control, or
FDP_IFC.1 Subset Information Flow Control]
FTP_ITC.1 Inter-TSF Trusted Channel

# Class FIA

# Identification and Authentication

243 Families in this class address the requirements for functions to establish and verify a claimed user identity.

244 Identification and Authentication is required to ensure that users are associated with the proper Security Attributes (e.g., identity, groups, roles, security or integrity levels).

245 The unambiguous identification of authorised users and the correct association of security attributes with users and subjects is critical to the enforcement of the intended security policies. The families in this class deal with determining and verifying the identity of users, determining their authority to interact with the TOE, and with the correct association of security attributes for each authorised user. Other classes of requirements (e.g., User Data Protection, Security Audit) are dependent upon correct identification and authentication of users in order to be effective.

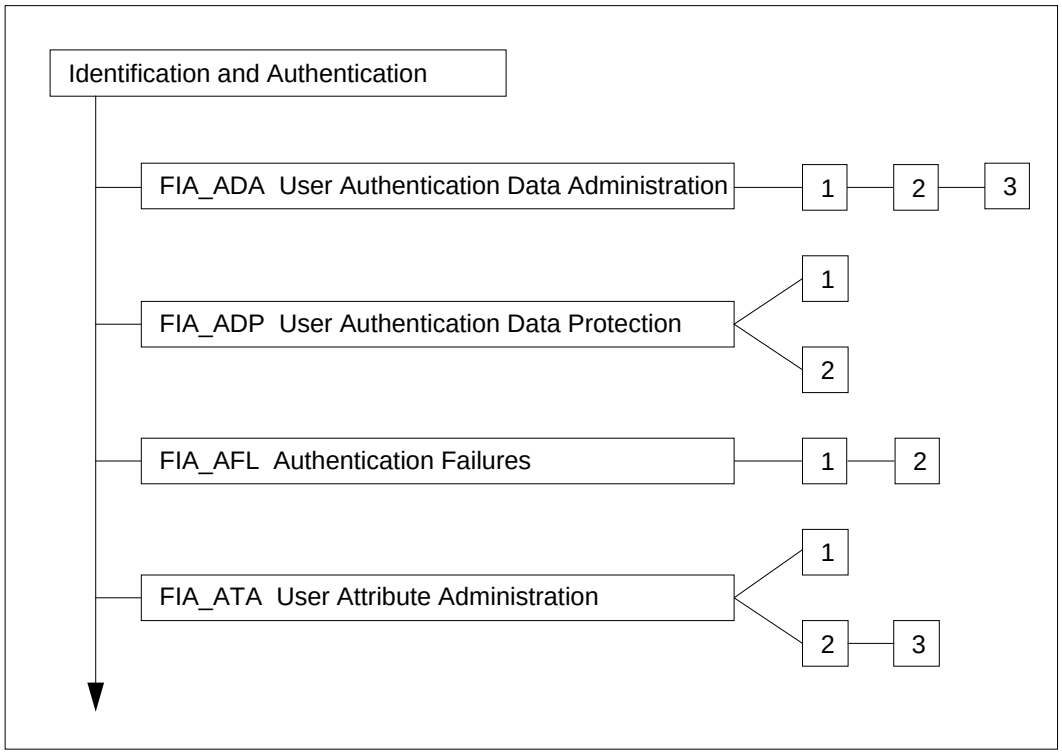



**Figure 2.10 - Identification and Authentication class decomposition**

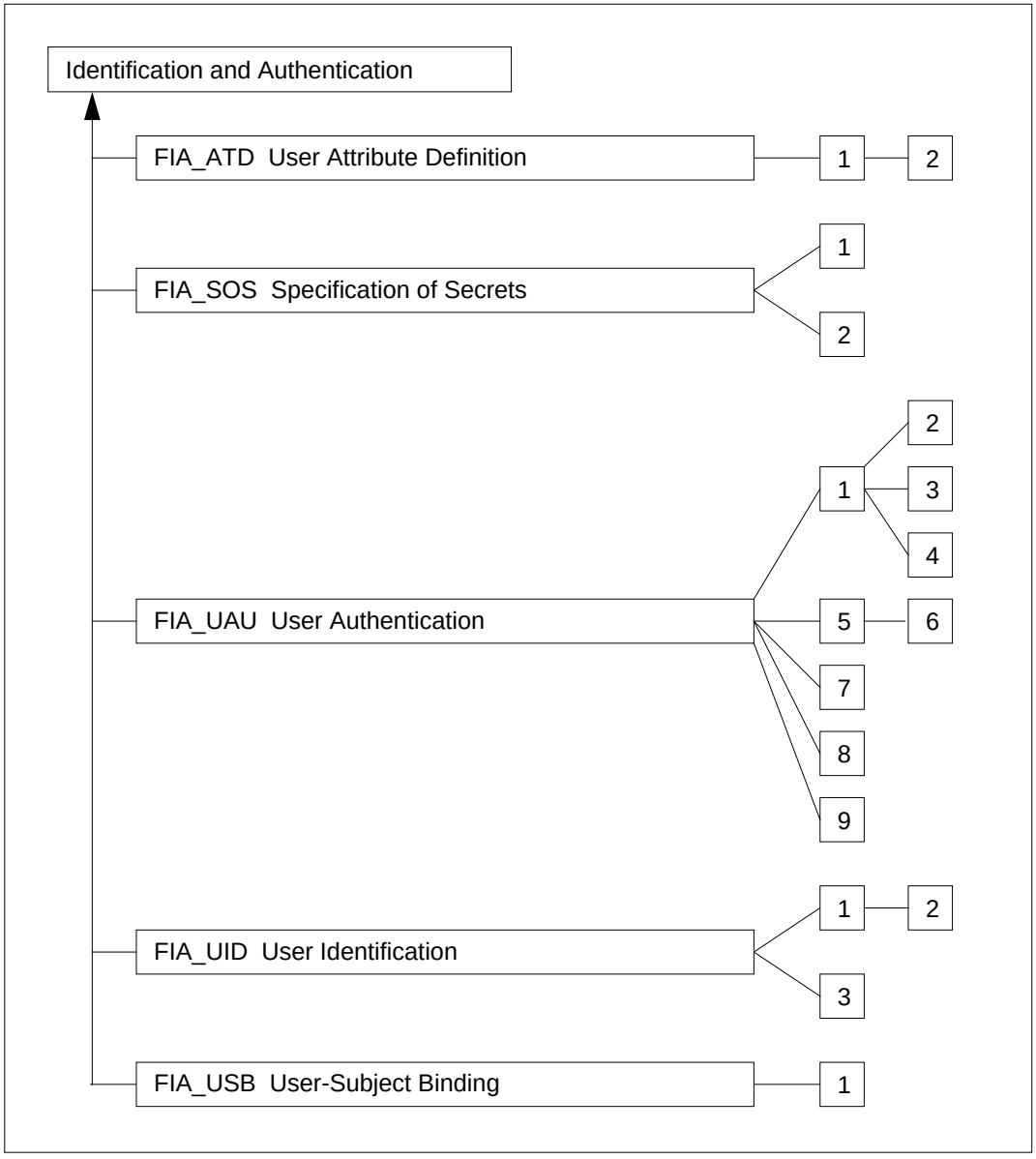


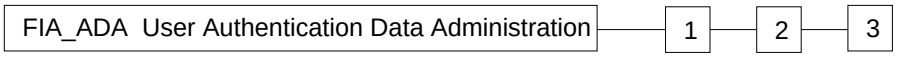

**Figure 2.11 - Identification and Authentication class decomposition (Cont.)**

# FIA_ADA User Authentication Data Administration

Family behaviour

246 This family defines requirements to initially set up or change user authentication data.

Component levelling




247 FIA_ADA.1 User Authentication Data Initialisation, requires that the TSF provide functions to initialise user authentication data related to the authentication mechanisms used.

248 FIA_ADA.2 Basic User Authentication Data Administration, requires that the TSF provide functions to initialise and modify user authentication data related to the authentication mechanisms used.

249 FIA_ADA.3 Expanded User Authentication Data Administration, requires that the TSF provide functions to initialise and modify any user's authentication data related to the authentication mechanisms used. It also requires the TSF to allow authorised users to modify their own authentication data.

Audit :

250 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: Successful use of any TSF authentication data management mechanisms.

b) Basic: All requests to use TSF authentication data management mechanisms.

## FIA_ADA.1 User Authentication Data Initialisation

Hierarchical to: no other components.

**FIA_ADA.1.1 The TSF shall provide functions for initialising user authentication data related to [assignment: *identified authentication mechanism*].**

**FIA_ADA.1.2 The TSF shall restrict the use of these functions to the authorised administrator.**

Dependencies : **FPT_TSA.1 Basic Security Administration**
**FIA_ADP.1 Basic User Authentication Data Protection**
**FIA_UAU.1 Basic User Authentication**

### FIA_ADA.2 Basic User Authentication Data Administration

Hierarchical to: FIA_ADA.1

**FIA_ADA.2.1** The TSF shall provide functions for initialising **and modifying** user authentication data related to [assignment: *identified authentication mechanism*].

**FIA_ADA.2.2** The TSF shall restrict the use of these functions to the authorised administrator.

Dependencies : FPT_TSA.1 Basic Security Administration
FIA_ADP.1 Basic User Authentication Data Protection
FIA_UAU.1 Basic User Authentication

### FIA_ADA.3 Expanded User Authentication Data Administration

Hierarchical to: FIA_ADA.2

**FIA_ADA.3.1** The TSF shall provide functions for initialising and modifying user authentication data related to [assignment: *identified authentication mechanism*].

**FIA_ADA.3.2** The TSF shall restrict the use of these functions **on the user authentication data for any user** to the authorised administrator.

**FIA_ADA.3.3** **The TSF shall allow authorised users to use these functions to modify their own authentication data in accordance with the TSP.**

Dependencies : FPT_TSA.1 Basic Security Administration
FIA_ADP.1 Basic User Authentication Data Protection
FIA_UAU.1 Basic User Authentication

# FIA_ADP User Authentication Data Protection

Family behaviour

251 This family defines the requirements to protect the user authentication data against unauthorised access or modification. It includes requirements to ensure the integrity of, or prevent the unauthorised use of, authentication data.

Component levelling


252 FIA_ADP.1 Basic User Authentication Data Protection, requires that the TSF provide protection of the authentication data that is permanently stored on the TOE.

253 FIA_ADP.2 Extended User Authentication Data Protection, requires that the TSF provide additional protection of raw authentication data while it is in the TOE.

Audit :

254 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: Successful requests to access user authentication data.

b) Basic: All requests to access user authentication data.

## FIA_ADP.1 Basic User Authentication Data Protection

Hierarchical to: no other components.

**FIA_ADP.1.1 The TSF shall protect from unauthorised observation, modification, and destruction authentication data that is stored in the TOE.**

Dependencies : **FIA_UAU.1 Basic User Authentication**

## FIA_ADP.2 Extended User Authentication Data Protection

Hierarchical to: no other components.

**FIA_ADP.2.1 The TSF shall protect from unauthorised observation, modification, and destruction the raw form of authentication data at all times while it resides in the TOE.**

Dependencies : **FIA_UAU.1 Basic User Authentication**

# FIA_AFL Authentication Failures

Family behaviour

255 This family contains requirements for defining values for some number of unsuccessful authentication attempts and TSF actions in cases of authentication attempt failures. Parameters include, but are not limited to, the number of failed authentication attempts and time thresholds.

Component levelling


256 FIA_AFL.1 requires that the TSF be able to terminate the session establishment process after a specified number of unsuccessful user authentication attempts. It also requires that, after termination of the session establishment process, the TSF be able to disable the user account or the point of entry (e.g., workstation) from which the attempts were made until an administrator-defined condition occurs.

257 FIA_AFL.2 requires the TSF to provide the authorised administrator with the capability to specify action to be taken on authentication failure.

## FIA_AFL.1 Basic Authentication Failure Handling

Hierarchical to: no other components.

**FIA_AFL.1.1** **The TSF shall be able to terminate the user session establishment process after [assignment: *number*] unsuccessful authentication attempts.**

**FIA_AFL.1.2** **After the termination of a user session establishment process, the TSF shall be able to disable the [selection: *user account, point of entry*] until [assignment: *conditions for re-enabling the user session establishment process*].**

Dependencies : **FIA_UAU.1 Basic User Authentication**

## FIA_AFL.2 Administrator Controlled Authentication Failure Handling

Hierarchical to: FIA_AFL.1

**FIA_AFL.2.1** The TSF shall be able to terminate the user session establishment process after [assignment: *number*] unsuccessful authentication attempts.

**FIA_AFL.2.2** After the termination of the session establishment process, the TSF shall **provide the authorised administrator with the ability to specify whether the [selection: *user account, point of entry*] is to be disabled** until [assignment: *conditions for re-enabling the user session establishment process*]**.**

Dependencies : FIA_UAU.1 Basic User Authentication

# FIA_ATA User Attribute Administration

Family behaviour

258 All authorised users have a set of attributes to support the enforcement of the TSP. This family defines the requirements to initially set up, change, or review this set of user attributes. This family defines requirements to enable new user identities to be added, and old user identities to be removed, modified, or invalidated.

Component levelling


259 FIA_ATA.1 User Attribute Initialisation, requires only that the TSF provide the ability to initialise user attributes.

260 FIA_ATA.2 Basic User Attribute Administration, requires that the TSF provide the ability for the authorised administrator to query and modify user attributes.

261 FIA_ATA.3 Extended User Attribute Administration, allows users to manage their own attributes in accordance with the TSP.

Audit :

262 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: Successful use of the user attribute administration functions.

b) Basic: All requests to use the user attribute administration functions.

c) Basic: Identification of the user attributes that have been modified.

d) Detailed: With the exception of specific sensitive attribute data items (e.g., authentication secrets, cryptographic keys), the new values of the attributes must be captured.

## FIA_ATA.1 User Attribute Initialisation

Hierarchical to: no other components.

FIA_ATA.1.1 **The TSF shall provide the ability to initialise user attributes with provided default values.**

Dependencies : **FIA_ATD.1 User Attribute Definition**
**FPT_TSA.1 Basic Security Administration**

## FIA_ATA.2 Basic User Attribute Administration

Hierarchical to: no other components.

**FIA_ATA.2.1** **The TSF shall provide the ability to [selection: *display, modify*] user attributes.**

**FIA_ATA.2.2** **The TSF shall limit the ability to modify user attributes to only the authorised administrator.**

Dependencies : **FIA_ATD.1 User Attribute Definition**
**FPT_TSA.1 Basic Security Administration**

## FIA_ATA.3 Extended User Attribute Administration

Hierarchical to: FIA_ATA.2

**FIA_ATA.3.1** The TSF shall provide the ability to [selection: *display*, *modify*] user attributes.

**FIA_ATA.3.2** The TSF shall limit the ability to modify **any** user**'s** attributes to only the authorised administrator.

**FIA_ATA.3.3** **The TSF shall allow users to modify their own attributes in accordance with the TSP.**

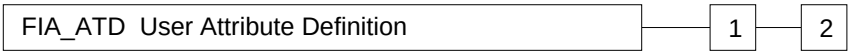
Dependencies : FIA_ATD.1 User Attribute Definition
FPT_TSA.1 Basic Security Administration

# FIA_ATD User Attribute Definition

Family behaviour

263 All authorised users may have a set of security attributes, other than the user's identity, that is used to enforce the TSP. This family defines the requirements for associating user security attributes with users as needed to support the TSP.

Component levelling




264 FIA_ATD.1 User Attribute Definition, allows user security attributes to be associated with groups of users.

265 FIA_ATD.2 Unique User Attribute Definition, requires that user security attributes be uniquely associated with each individual user.

## FIA_ATD.1 User Attribute Definition

Hierarchical to: no other components.

**FIA_ATD.1.1** **The TSF shall provide, for each user, a set of security attributes necessary to enforce the TSP.**

Dependencies : **ADV_FSP.1 TOE and security policy**

## FIA_ATD.2 Unique User Attribute Definition

Hierarchical to: FIA_ATD.1

**FIA_ATD.2.1** The TSF shall provide, for each user, a **unique** set of security attributes necessary to enforce the TSP.

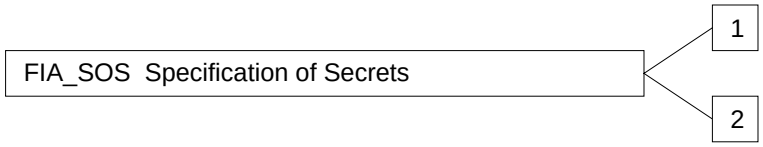
Dependencies : ADV_FSP.1 TOE and security policy

# FIA_SOS Specification of Secrets

Family behaviour

266 This family defines requirements for mechanisms that enforce defined quality metrics on provided secrets and generate secrets to satisfy the defined metric.

Component levelling




267 FIA_SOS.1 Selection of Secrets requires the TSF to verify that secrets meet defined quality metrics.

268 FIA_SOS.2 TSF Generation of Secrets requires the TSF to be able to generate secrets that meet defined quality metrics.

Audit :

269 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: Rejection by the TSF of any tested secret.

b) Basic: Rejection or acceptance by the TSF of any tested secret.

c) Detailed: Identification of any changes to the defined quality metrics.

## FIA_SOS.1 Selection of Secrets

Hierarchical to: no other components.

**FIA_SOS.1.1 The TSF shall provide a mechanism to verify that secrets meet [assignment: *a defined quality metric*].**

Dependencies : No dependencies.

## FIA_SOS.2 TSF Generation of Secrets

Hierarchical to: no other components.

**FIA_SOS.2.1 The TSF shall provide a mechanism to generate secrets that meet [assignment: *a defined quality metric*].**

**FIA_SOS.2.2** **The TSF shall be able to enforce the use of TSF generated secrets for [assignment: *list of TSF functions*].**

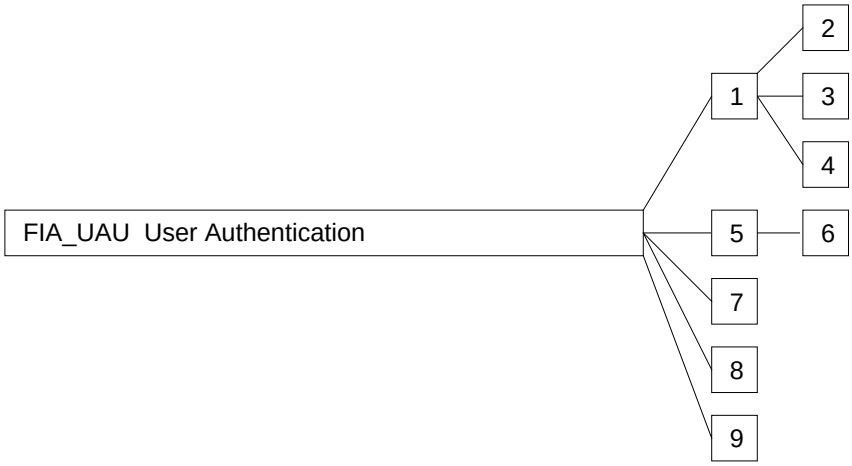
Dependencies : No dependencies.

# FIA_UAU User Authentication

## Family behaviour

270 This family defines the types of user authentication mechanisms supported by the TSF. This family also defines the required attributes on which the user authentication mechanisms must be based.

## Component levelling




271 FIA_UAU.1 Basic User Authentication, includes only minimal forms of individual user authentication, and is intended for use in products that will have limited exposure to authentication-based attacks.

272 FIA_UAU.2 Single-use Authentication Mechanisms, requires an authentication mechanism that operates with single-use authentication data.

273 FIA_UAU.3 Integrity of Authentication, requires the authentication mechanism to be able to detect and prevent the use of authentication data that has been forged or copied.

274 FIA_UAU.4 Multiple Authentication Mechanisms, requires that two or more different authentication mechanisms be provided and used to authenticate user identities.

275 FIA_UAU.5 Policy-based Authentication Mechanisms, requires the ability to specify separate authentication mechanisms for specific authentication events.

276 FIA_UAU.6 Configurable Authentication Mechanisms, allows a user authorised to perform administrative functions to specify separate authentication mechanisms for specific authentication events.

277 FIA_UAU.7 On-demand Authentication, requires that the TSF be able to re-authenticate the user at times after initial authentication.

278 FIA_UAU.8 Timing of Authentication, allows a user to perform certain functions prior to the authentication of the user's identity.

279 FIA_UAU.9 Installable Authentication Mechanisms, requires an interface to support the installation of new authentication mechanisms.

Audit :

280 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: Successful use of the authentication mechanism.

b) Basic: Any use of the authentication mechanism.

Audit : for FIA_UAU.6

281 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: Audit the action of configuring the mapping of authentication mechanisms to specific authentication events.

Audit : for FIA_UAU.9

282 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: Installation of an authentication mechanism.

## FIA_UAU.1 Basic User Authentication

Hierarchical to: no other components.

**FIA_UAU.1.1** **The TSF shall authenticate any user's claimed identity prior to performing any functions for the user.**

Dependencies : **FIA_UID.1 Basic User Identification**
**FIA_ADA.1 User Authentication Data Initialisation**

## FIA_UAU.2 Single-use Authentication Mechanisms

Hierarchical to: FIA_UAU.1

**FIA_UAU.2.1** The TSF shall authenticate any user's claimed identity prior to performing any functions for the user.

**FIA_UAU.2.2** **The TSF shall prevent reuse of authentication data related to [assignment: *identified authentication mechanisms*].**

Dependencies : FIA_UID.1 Basic User Identification
FIA_ADA.1 User Authentication Data Initialisation

### FIA_UAU.3 Integrity of Authentication

Hierarchical to: FIA_UAU.1

**FIA_UAU.3.1** The TSF shall authenticate any user's claimed identity prior to performing any functions for the user.

**FIA_UAU.3.2** **The TSF shall detect and prevent use of authentication data that has been forged by any user of the TSF.**

**FIA_UAU.3.3** **The TSF shall detect and prevent use of authentication data that has been copied from any other user of the TSF.**

Dependencies : FIA_UID.1 Basic User Identification
FIA_ADA.1 User Authentication Data Initialisation

### FIA_UAU.4 Multiple Authentication Mechanisms

Hierarchical to: FIA_UAU.1

**FIA_UAU.4.1** The TSF shall **provide [assignment: *number*] different mechanisms [assignment: *list of different mechanisms*] to** authenticate any user's claimed identity prior to performing any functions for the user**.**

**FIA_UAU.4.2** **The TSF shall, by default, use all of these mechanisms to authenticate any user's claimed identity, with authentication being successful if and only if all mechanisms individually indicate successful authentication.**

Dependencies : FIA_UID.1 Basic User Identification
FIA_ADA.1 User Authentication Data Initialisation

### FIA_UAU.5 Policy-based Authentication Mechanisms

Hierarchical to: no other components.

**FIA_UAU.5.1** **The TSF shall provide [assignment: *number*] different mechanisms [assignment: *list of different mechanisms*] to authenticate any user's claimed identity.**

**FIA_UAU.5.2** **The TSF shall enforce the use of [refinement: *separate authentication mechanisms for specific authentication events*], with authentication being successful if and only if all of the defined mechanisms individually indicate successful authentication.**

Dependencies : **FIA_UAU.1 Basic User Authentication**

### FIA_UAU.6 Configurable Authentication Mechanisms

Hierarchical to: FIA_UAU.5

**FIA_UAU.6.1** The TSF shall provide [assignment: *number*] different mechanisms [assignment: *list of different mechanisms*] to authenticate any user's claimed identity.

**FIA_UAU.6.2** The TSF shall enforce the use of [refinement: *separate authentication mechanisms for specific authentication events*], with authentication being successful if and only if all of the defined mechanisms individually indicate successful authentication.

**FIA_UAU.6.3** **The TSF shall allow the authorised administrator to associate [refinement: *separate authentication mechanisms with specific authentication events*]**.

Dependencies : FIA_UAU.1 Basic User Authentication

### FIA_UAU.7 On-demand Authentication

Hierarchical to: no other components.

**FIA_UAU.7.1** **The TSF shall re-authenticate users under the following circumstances: [assignment: *list of conditions requiring re-authentication*].**

Dependencies : **FIA_UAU.1 Basic User Authentication**

### FIA_UAU.8 Timing of Authentication

Hierarchical to: no other components.

**FIA_UAU.8.1** **The TSF shall allow users to perform [assignment: *list of TSF-mediated actions*] before the user's claimed identity is authenticated.**

**FIA_UAU.8.2** **The TSF shall perform the authentication of any user's claimed identity before performing any other TSF-mediated actions on behalf of the user.**

Dependencies : **FIA_UAU.1 Basic User Authentication**

### FIA_UAU.9 Installable Authentication Mechanisms

Hierarchical to: no other components.

**FIA_UAU.9.1** **The TSF shall provide the ability for the authorised administrator to incorporate installable authentication mechanisms into the TSF.**

**FIA_UAU.9.2** **The TSF shall use the installed authentication mechanism [selection: *in place of, in addition to*] any existing authentication mechanism.**

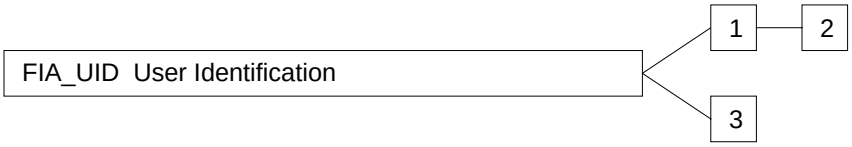
Dependencies : **FIA_UID.1 Basic User Identification**
**FPT_TSA.1 Basic Security Administration**

# FIA_UID User Identification

Family behaviour

283 This family defines the conditions under which users shall be required to identify themselves before performing any other actions that are to be mediated by the TSF and which require user identification.

Component levelling




284 FIA_UID.1 Basic User Identification, allows users to share the same user identity.

285 FIA_UID.2 Unique Identification of Users, requires that each user have a unique identity.

286 FIA_UID.3 Timing of Identification, allows users to perform certain actions before being identified by the TSF.

Audit :

287 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: Successful use of the user identification mechanism, including the user identity provided.

b) Basic: All attempts to use the user identification mechanism, including the user identity provided.

## FIA_UID.1 Basic User Identification

Hierarchical to: no other components.

**FIA_UID.1.1 The TSF shall identify each user before performing any actions requested by the user.**

Dependencies : **FIA_ATD.1 User Attribute Definition**

### FIA_UID.2 Unique Identification of Users

Hierarchical to: FIA_UID.1

**FIA_UID.2.1** The TSF shall **uniquely** identify each user before performing any actions requested by the user.

Dependencies : **FIA_ATD.2 Unique User Attribute Definition**

### FIA_UID.3 Timing of Identification

Hierarchical to: no other components.

**FIA_UID.3.1** **The TSF shall allow users to perform [assignment: *list of actions*] before identifying the user.**

**FIA_UID.3.2** **The TSF shall identify each user before performing any other actions on behalf of the user.**

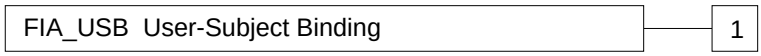
Dependencies : No dependencies.

# FIA_USB User-Subject Binding

Family behaviour

288 An authenticated user, to perform functions in the TOE, typically activates a subject. The user's security attributes are associated (totally or partially) with this subject. This family defines requirements to create and maintain the association of the user's security attributes to a subject acting on the user's behalf.

Component levelling




289 FIA_USB.1 User-Subject Binding requires the maintenance of an association between the user's security attributes and a subject acting on the user's behalf.

Audit :

290 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: Successful binding of user security attributes to a subject (e.g., creation of a subject).

b) Basic: Success and failure of binding of user security attributes to a subject (e.g., success and failure to create a subject).

## FIA_USB.1 User-Subject Binding

Hierarchical to: no other components.

**FIA_USB.1.1 The TSF shall associate the appropriate user security attributes with subjects acting on behalf of that user.**

Dependencies : **FIA_ATD.1 User Attribute Definition**
**ADV_FSP.1 TOE and security policy**
**FDP_ACI.1 Static Attribute Initialisation**

# Class FPR

# Privacy

291 This class contains privacy requirements. These requirements provide a user protection against discovery and misuse of his identity by other users.

292 This class is based on the current available knowledge about privacy techniques. Since research in this area is still ongoing, in the future these components might need expansion or revision.

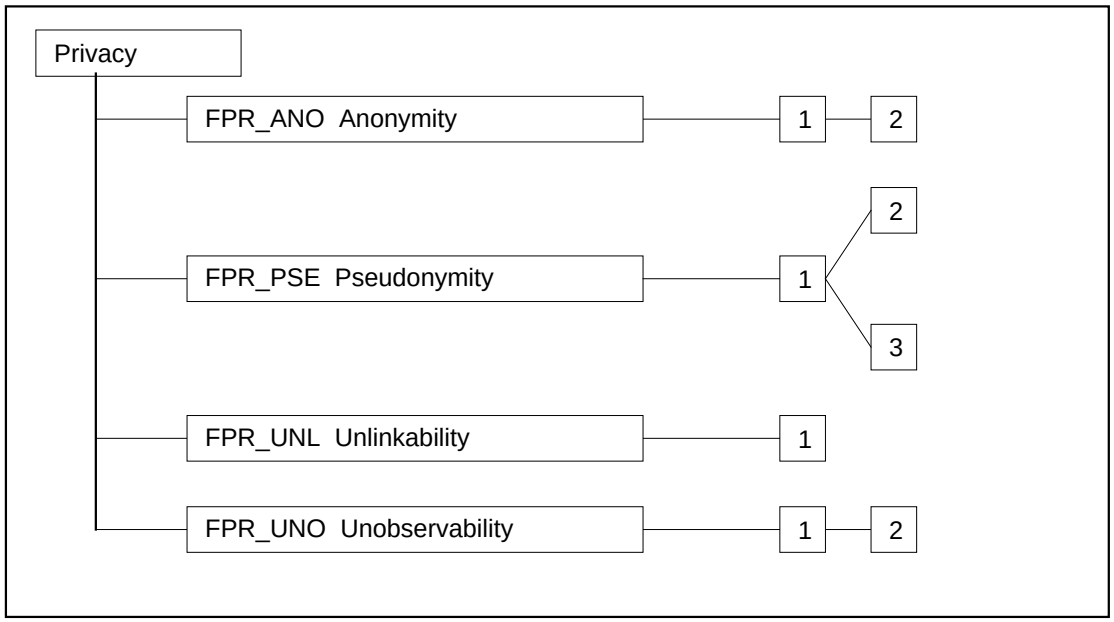



**Figure 2.12 - Privacy class decomposition**

# FPR_ANO Anonymity

Family behaviour

293 This family ensures that a user may use a resource or service without disclosing the user's identity. The requirements for Anonymity provide protection of the user identity. Anonymity is not intended to protect the subject identity.

Component levelling


294 FPR_ANO.1 Anonymity requires that other users or subjects are unable to determine the identity of a user bound to a subject or operation.

295 FPR_ANO.2 TSF Anonymity enhances the requirements of FPR_ANO.1 by ensuring that the TSF does not ask for the user identity.

## FPR_ANO.1 Anonymity

Hierarchical to: no other components.

**FPR_ANO.1.1** **The TSF shall ensure that [assignment: *set of users and/or subjects* ] working together, [selection: *including, excluding*] authorised administrators, are unable to determine the user identity bound to [assignment: *list of subjects and/ or operations*].**

Dependencies : No dependencies.

## FPR_ANO.2 TSF Anonymity

Hierarchical to: FPR_ANO.1

**FPR_ANO.2.1** The TSF shall ensure that [assignment: *set of users and/or subjects*] working together, [selection: *including, excluding*] authorised administrators, are unable to determine the user identity bound to [assignment: *list of subjects and/or operations*].

**FPR_ANO.2.2** **The TSF shall not solicit any reference to the user identity in order to initiate actions on behalf of [assignment: *list of subjects*] or subjects requesting [assignment: *list of operations*].**

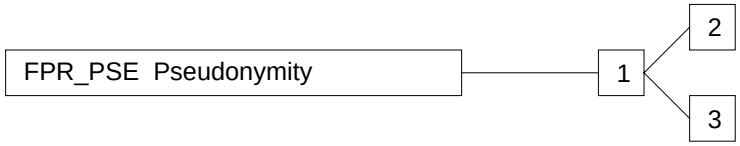
Dependencies : No dependencies.

# FPR_PSE Pseudonymity

Family behaviour

296 This family ensures that a user may use a resource or service without disclosing its user identity, but can still be accountable for that use.

Component levelling




297 FPR_PSE.1 Pseudonymity requires that a set of users and/or subjects are unable to determine the identity of a user bound to a subject or operation, but that this user is still accountable for its actions.

298 FPR_PSE.2 Reversible Pseudonymity requires the TSF to provide a capability to determine the original user identity based on a provided alias.

299 FPR_PSE.3 Alias Pseudonymity requires the TSF to follow certain construction rules for the alias to the user identity.

Audit : for FPR_PSE.2

300 The following actions shall be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: The subject /user which requested resolution of the user identity should be audited.

## FPR_PSE.1 Pseudonymity

Hierarchical to: no other components.

**FPR_PSE.1.1** **The TSF shall ensure that [assignment: *set of users and/or subjects*] working together, [selection: *including, excluding*] authorised administrators, are unable to determine the user identity bound to [assignment: *list of subjects and/ or operations*].**

**FPR_PSE.1.2** **The TSF shall be able to provide an alias to the user identity related to the above mentioned lists of subjects or operations.**

Dependencies : No dependencies.

### FPR_PSE.2 Reversible Pseudonymity

Hierarchical to: FPR_PSE.1

**FPR_PSE.2.1** The TSF shall ensure that [assignment: *set of users and/or subjects*] working together, [selection: *including, excludin*g] authorised administrators, are unable to determine the user identity bound to [assignment: *list of subjects and/or operations*].

**FPR_PSE.2.2** The TSF shall be able to provide an alias to the user identity related to the above mentioned lists of subjects or operations.

**FPR_PSE.2.3** **The TSF shall provide [selection: *an authorised administrator,* [assignment*: list of trusted subjects*]] a capability to determine the user identity based on the provided alias only under the conditions [assignment: *list of conditions*].**

Dependencies : **FIA_UID.1 Basic User Identification**

### FPR_PSE.3 Alias Pseudonymity

Hierarchical to: FPR_PSE.1

**FPR_PSE.3.1** The TSF shall ensure that [assignment: *set of users and/or subjects*] working together, [selection: *including, excludin*g] authorised administrators, are unable to determine the user identity bound to [assignment: *list of subjects and/or operations*].

**FPR_PSE.3.2** The TSF shall be able to provide an alias to the user identity related to the above mentioned lists of subjects or operations.

**FPR_PSE.3.3** **The TSF shall provide an alias to the user identity which shall be identical to an alias provided previously under the following conditions: [assignment: *list of conditions*] and unrelated to previously provided aliases otherwise.**

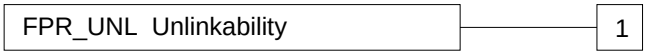
Dependencies : No dependencies.

# FPR_UNL Unlinkability

Family behaviour

301 This family ensures that a user may make multiple uses of resources or services without others being able to link these uses together.

Component levelling




302 FPR_UNL.1 Unlinkability requires that users and/or subjects are unable to determine whether the same user caused certain specific operations in the system.

## FPR_UNL.1 Unlinkability

Hierarchical to: no other components.

**FPR_UNL.1.1** **The TSF shall ensure that [assignment: *set of users and/or subjects*] working together, [selection: *including, excluding*] authorised administrators, are unable to determine whether [assignment: *list of operations*] [selection: '*were caused by the same user', 'are related as follows* [assignment: *list of relations*]'].**

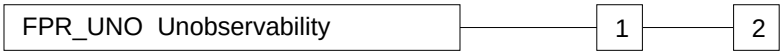
Dependencies : No dependencies.

# FPR_UNO Unobservability

Family behaviour:

303 This family ensures that a subject may use a resource or service without others, especially third parties, being able to observe that the resource or service is being used.

Component levelling




304 FPR_UNO.1 Unobservability requires that users and/or subjects cannot determine whether an object is being used.

305 FPR_UNO.2 Authorised Administrator Observability requires the TSF to provide the authorised administrator with a capability to observe the use of a resource or service.

## FPR_UNO.1 Unobservability

Hierarchical to: no other components.

**FPR_UNO.1.1** **The TSF shall ensure that [assignment: *set of users and/or subjects*] working together, [selection: *including, excluding*] authorised administrators, are unable to observe the operation [assignment: *list of operations*] on [assignment: *list of objects*] by another user or subject.**

Dependencies : No dependencies.

## FPR_UNO.2 Authorised Administrator Observability

Hierarchical to: FPR_UNO.1

**FPR_UNO.2.1** The TSF shall ensure that [assignment: *set of users and/or subjects*] working together, [selection: *including, excluding*] authorised administrators, are unable to observe the operation [assignment: *list of operations*] on [assignment: *list of objects*] by another user or subject.

**FPR_UNO.2.2** **The TSF shall provide an authorised administrator with the capability to determine the utilisation of resources and/or services.**

Dependencies : No dependencies.

# Class FPT

# Protection of the Trusted Security Functions

306 This class contains families of functional requirements that relate to the integrity and management of the mechanisms that provide the TSF (independent of TSP-specifics) and to the integrity and management of TSF data (independent of the specific contents of the TSP data). In some sense, families in this class may appear to duplicate components in the FDP (User Data Protection) class; they may even be implemented using the same mechanisms. However, FDP focuses on user data protection, while FPT focuses on TSF data protection. In fact, components from the FPT class are necessary even in the absence of any user data protection, to provide confidence in the enforcement of other policies (such as accountability) that may be specified in the PP/ST.

307 From the point of view of this class, there are three significant portions that make up the TSF:

a) The TSF's *abstract machine*, which is the virtual or physical machine upon which the specific TSF software under evaluation executes.

b) The TSF's *software*, which executes on the abstract machine and implements the mechanisms that enforce the TSP.

c) The TSF's *data*, which are the administrative databases that guide the enforcement of the TSP.

308 All of the families in the FPT class can be related to these three areas, and fall into the following groupings:

a) Families that address protection of the TSF mechanisms. These families are:

1) FPT_PHP (TSF Physical Protection) and FPT_SWM (TSF Software Modification), which provide the authorised administrator with the ability to detect external attacks on the parts of the TOE that comprise the TSF.

2) FPT_AMT (Underlying Abstract Machine Test) and FPT_TST (TSF Self Test), which provide the authorised administrator with the ability to verify the correct operation of the TSF and the integrity of the TSF data and underlying abstract machine.

3) FPT_SEP (Domain Separation) and FPT_RVM (Reference Mediation), which protect the TSF during execution and ensure that the TSF cannot be bypassed. When appropriate components from these families are combined with the appropriate components from ADV_INT (TSF internals), the TOE can be said to have what has been traditionally called a “Reference Monitor.” The Reference Monitor is that portion of the TSF responsible for the enforcement of the TSP; it has the following three characteristics:

- Untrusted subjects cannot interfere with its operation; i.e., it is tamperproof. This is addressed by the components in the FPT_SEP family.

- Untrusted subjects cannot bypass its checks; i.e., it is always invoked. This is addressed by the components in the FPT_RVM family.

- It is simple enough to be analysed and its behaviour understood (i.e., its design is conceptually simple.) This is addressed by the components in the ADV_INT family.

4) FPT_RCV (Trusted Recovery), FPT_FLS Fail Secure, and FPT_TRC (Internal TOE TSF Data Replication Consistency), which address the behaviour of the TSF when failure occurs and immediately after.

5) FPT_ITA (Inter-TSF Availability of TSF Data), FPT_ITC (Inter-TSF Confidentiality of TSF Data), FPT_ITI (Inter-TSF Integrity of TSF Data), which address the protection and availability of TSF data between the TSF and a remote TSF. FPT_ITT (Internal TOE TSF Data Transfer) is similar to the previous three families, but addresses protection of TSF data when it is transmitted between parts of the TOE.

6) FPT_RPL (Replay Detection and Prevention), which addresses the replay of various types of information and/or operations.

7) FPT_SSP State Synchrony Protocol, which addresses the state synchrony required between two TSF components.

8) FPT_STM (Time Stamps), which addresses timing consistency internal to the TSF.

b) Families that address the management of TSF data. These families are:

1) FPT_SAE Security Attribute Expiration, which addresses the expiration of the validity of security attributes.

2) FPT_REV Revocation, which address the revocation of security attributes.

3) FPT_TDC Inter-TSF TSF Data Consistency, which addresses the consistency of TSF data shared between TSF of distinct TOEs.

4) FPT_TSA (TOE Security Administration), which addresses the functions that must be available to the administrator that are independent of those related to any other class.

5) FPT_TSM (TOE Security Management), which addresses how management of the TSF is structured.

6) FPT_TSU (TOE Administrative Safe Use), which addresses the ease of use of the administrative interface.

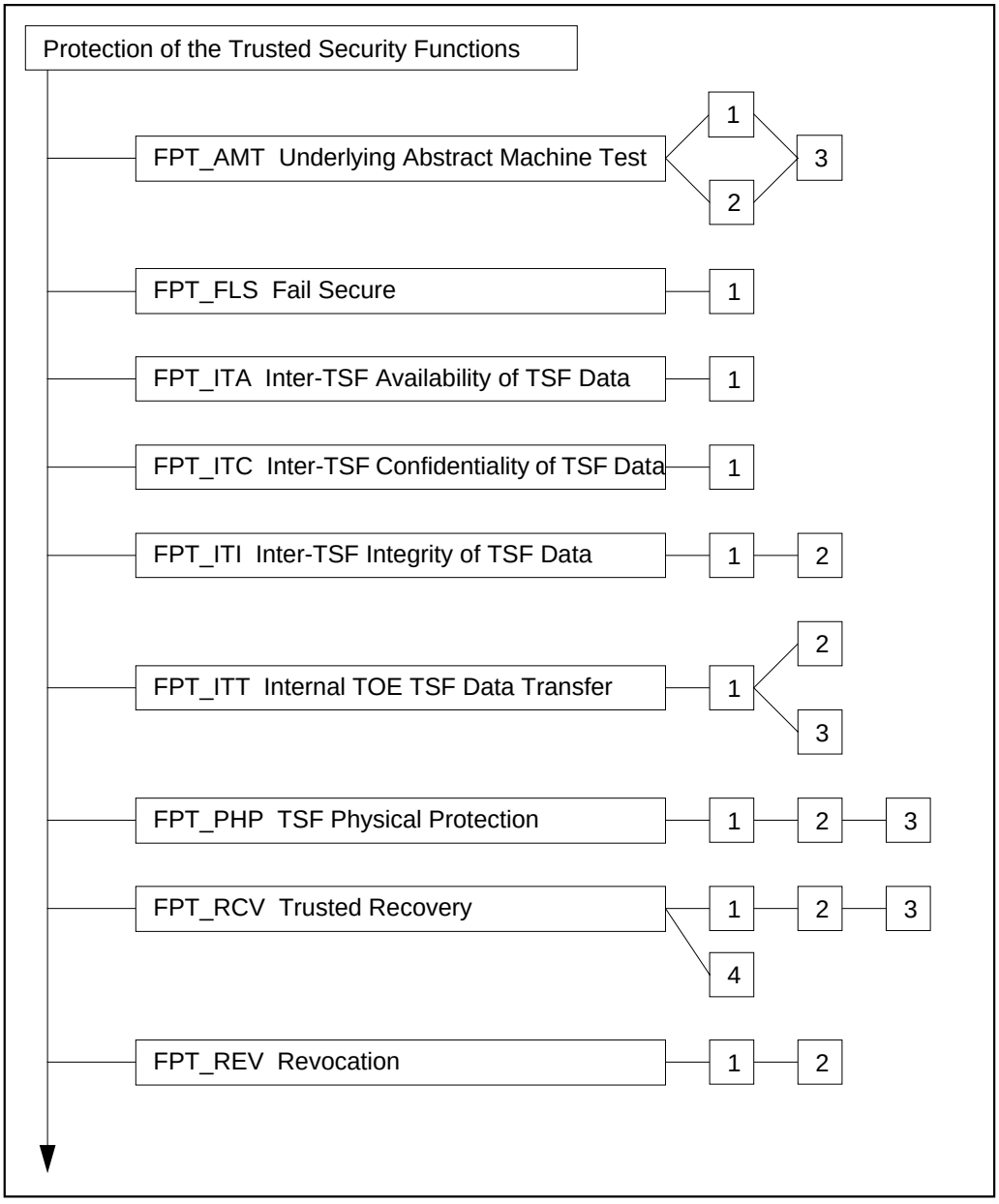



**Figure 2.13 - Protection of the Trusted Security Functions class decomposition**

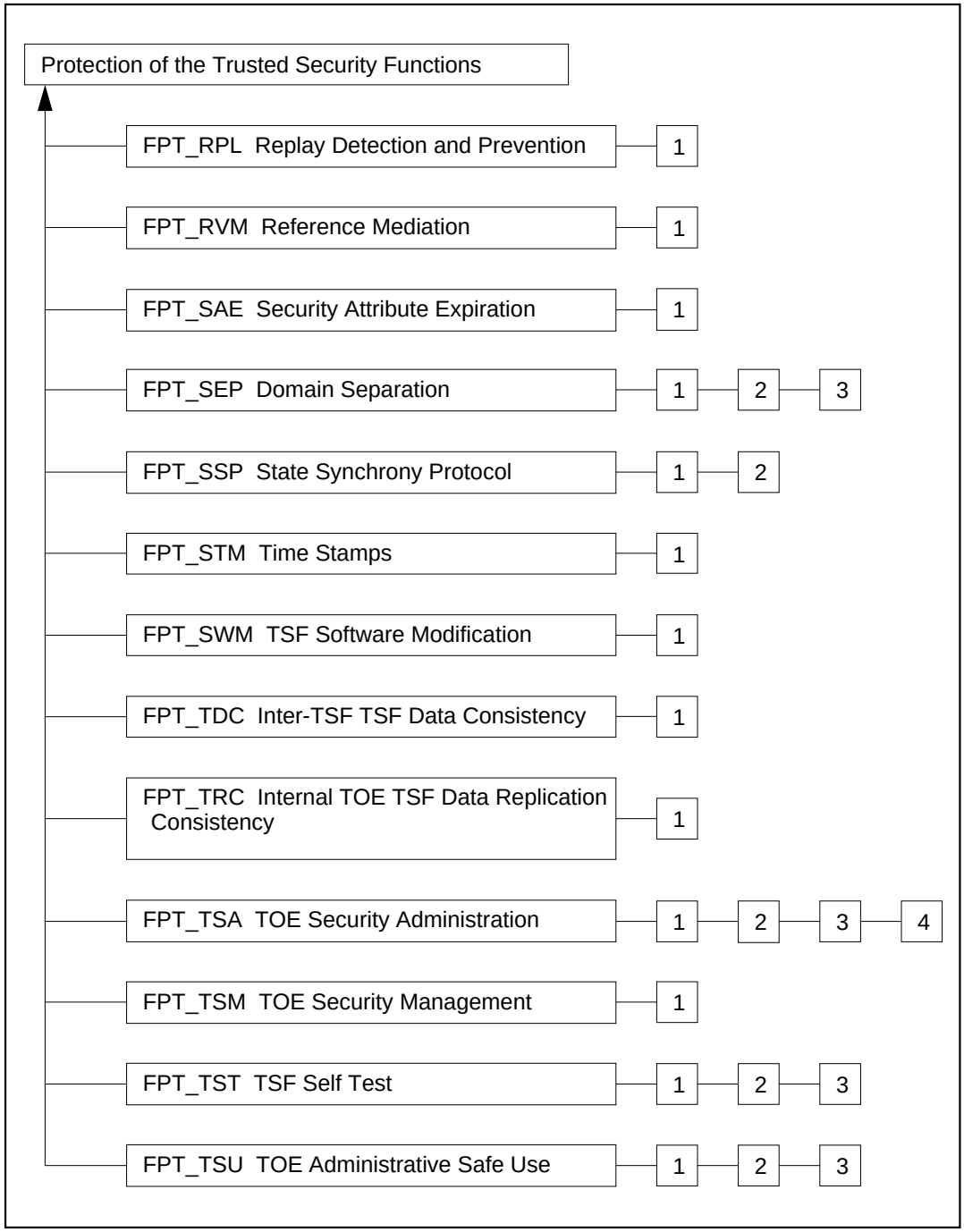


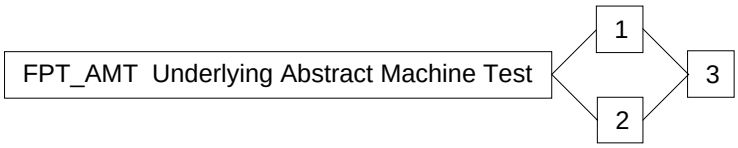

**Figure 2.14 - Protection of the Trusted Security Functions class decomposition (Cont.)**

# FPT_AMT Underlying Abstract Machine Test

## Family behaviour

309 This family defines requirements for the TSF to perform testing to demonstrate the security assumptions made about the underlying abstract machine upon which the TSF relies. This "abstract" machine could be a hardware/firmware platform, or it could be some known and assessed hardware/software combination acting as a virtual machine. Examples could be testing hardware page protection, sending sample packets across a network to ensure receipt, verifying the behaviour of the virtual machine interface, etc. These tests can be carried out either in some maintenance state, at start-up, on-line, or continuously. The actions to be taken by the TOE as the result of self testing are defined in FPT_RCV.

## Component levelling




310 FPT_AMT.1 Abstract Machine Testing, provides the authorised administrator the ability to test the underlying abstract machine. These tests may be performed during normal operation, in a maintenance mode, or off-line.

311 FPT_AMT.2 Abstract Machine Testing During Start-Up, provides for both human-invoked periodic tests and TSF-invoked testing during start-up.

312 FPT_AMT.3 Abstract Machine Testing During Normal Operation, provides for periodic testing of the underlying abstract machine by the TSF during normal operation, as well as human-invoked periodic tests and TSF-invoked testing during start-up.

## Audit :

313 The following actions should be audited if FAU_GEN Security Audit Data Generation is included in the PP/ST:

a) Basic: Execution of the tests of the underlying machine and the results of the tests.

### FPT_AMT.1 Abstract Machine Testing

Hierarchical to: no other components.

**FPT_AMT.1.1 The TSF shall provide the authorised administrator with the capability to demonstrate the correct operation of the security-relevant functions provided by the TSF's underlying abstract machine.**

Dependencies : No dependencies.

### FPT_AMT.2 Abstract Machine Testing During Start-Up

Hierarchical to: no other components.

**FPT_AMT.2.1** **The TSF shall run a suite of self tests during initial start-up in order to demonstrate the correct operation of the functions provided by the TSF's underlying abstract machine.**

Dependencies : No dependencies.

### FPT_AMT.3 Abstract Machine Testing During Normal Operation

Hierarchical to: FPT_AMT.1 and FPT_AMT.2

**FPT_AMT.3.1** The TSF shall provide the authorised administrator with the capability to demonstrate the correct operation of the security-relevant functions provided by the TSF's underlying abstract machine.

**FPT_AMT.3.2** The TSF shall run a suite of self tests during initial start-up **and periodically during normal operation** in order to demonstrate the correct operation of the functions provided by the TSF's underlying abstract machine.

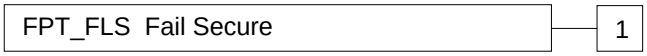
Dependencies : No dependencies.

## FPT_FLS Fail Secure

Family behaviour

314 The requirements of this family ensure that the TOE will not violate its TSP in the event of identified categories of failures in the TSF.

Component levelling




315 This family consists of only one component, FPT_FLS.1 Failure with Preservation of Secure State, which requires that the TSF maintain a secure state in the face of identified failures.

Audit :

316 Although it is desirable to audit situations in which failure with preservation of secure state occurs, it is not possible in all situations. The PP/ST author should specify those situations in which audit is desired and feasible.

### FPT_FLS.1 Failure with Preservation of Secure State

Hierarchical to: no other components.

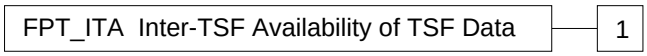
**FPT_FLS.1.1 The TSF shall preserve a secure state when [assignment: *list of types of TSF failures*] occur.**

Dependencies : **ADV_FSP.2 Informal security policy model**

# FPT_ITA Inter-TSF Availability of TSF Data

317 This family defines the rules for the prevention of loss of availability of TSF data moving between the TOE's TSF and the TSF of another TOE. This data could be TSF critical data such as passwords or keys, or it could be TSF executable code.

Component levelling




318 This family consists of only one component, FPT_ITA.1 Inter-TSF Availability Within a Defined Availability Factor, which requires that the TSF ensure, to an identified degree of probability, that TSF data made available between TSFs can be obtained by the receiving TSF.

## FPT_ITA.1 Inter-TSF Availability Within a Defined Availability Factor

Hierarchical to: no other components.

**FPT_ITA.1.1 The TSF shall ensure the availability of [assignment: *list of types of TSF data*] provided to a remote TSF within [refinement: *a defined availability metric*].**

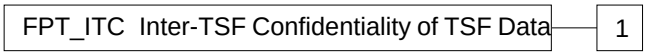
Dependencies : No dependencies.

# FPT_ITC Inter-TSF Confidentiality of TSF Data

Family behaviour

319 This family defines the rules for the protection from unauthorised disclosure of TSF data moving between the TOE's TSF and the TSF of another TOE. This data could be TSF critical data such as passwords, keys, audit records, or TSF executable code.

Component levelling




320 This family consists of only one component, FPT_ITC.1 Inter-TSF Confidentiality During Transmission, which requires that the TSF ensure that data transmitted between TSFs is protected from disclosure while in transit.

## FPT_ITC.1 Inter-TSF Confidentiality During Transmission

Hierarchical to: no other components.

**FPT_ITC.1.1 The TSF shall protect any TSF data transmitted from the TSF to a remote TSF from unauthorised disclosure.**

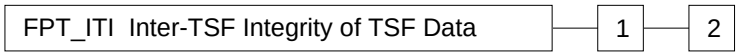
Dependencies : No dependencies.

# FPT_ITI Inter-TSF Integrity of TSF Data

Family behaviour

321 This family defines the rules for the protection, from unauthorised modification, of TSF data moving between the TOE's TSF and the TSF of another TOE. This data could, for example, be TSF critical data such as passwords, keys, audit data, or TSF executable code.

Component levelling




322 FPT_ITI.1 Inter-TSF Detection of Modification, provides the ability to detect modification of TSF data when it is transmitted to a remote TSF, under the assumption that the remote TSF is cognisant of the mechanism used.

323 FPT_ITI.2 Inter-TSF Detection and Correction of Modification, provides the ability for the remote TSF not only to detect modification, but to prevent (either through making the modification impossible, or by correcting the modification after it occurs) modification of transmitted TSF data, under the assumption that the remote TSF is cognisant of the mechanism used.

## FPT_ITI.1 Inter-TSF Detection of Modification

Hierarchical to: no other components.

**FPT_ITI.1.1** **The TSF shall provide the capability to detect modification within [refinement: *a defined modification metric*] of any TSF data transmitted from the TSF to a remote TSF.**

Dependencies : No dependencies.

## FPT_ITI.2 Inter-TSF Detection and Correction of Modification

Hierarchical to: FPT_ITI.1

**FPT_ITI.2.1** The TSF shall provide the capability to detect modification within [refinement: *a defined modification metric*] of any TSF data transmitted from the TSF to a remote TSF.

**FPT_ITI.2.2** **The TSF shall provide the capability to correct [assignment: *type of modification*] of any TSF data transmitted from the TSF to a remote TSF.**

Dependencies : No dependencies.

# FPT_ITT Internal TOE TSF Data Transfer

Family behaviour

324 This family provides requirements that address protection of TSF data when it is transferred between parts of a TOE across an internal channel.

Component levelling


325 FPT_ITT.1 Basic Internal TSF Data Transfer Protection, requires that TSF data be protected when transmitted between parts of the TOE.

326 FPT_ITT.2 TSF Data Transmission Separation by Attribute, requires that the TSF separate user data from TSF data during transmission.

327 FPT_ITT.3 TSF Data Integrity Monitoring, is distinct from the first two components, and requires that the TSF monitor user data transmitted between parts of the TOE for identified integrity errors.

## FPT_ITT.1 Basic Internal TSF Data Transfer Protection

Hierarchical to: no other components.

**FPT_ITT.1.1** **The TSF shall use [assignment: *specific mechanism*] to protect TSF data from [selection: *disclosure, modification, disclosure and modification*] when it is transmitted between physically-separated parts of the TOE.**

**FPT_ITT.1.2** **If the TSF provides multiple mechanisms to protect TSF data during transmission between physically-separated parts of the TOE, the TSF shall provide authorised administrators with the ability to select the method used.**

Dependencies : No dependencies.

## FPT_ITT.2 TSF Data Transmission Separation by Attribute

Hierarchical to: FPT_ITT.1

**FPT_ITT.2.1** The TSF shall use [assignment: *specific mechanism*] to protect TSF data from [selection: *disclosure, modification, disclosure and modification*] when it is transmitted between physically-separated parts of the TOE.

**FPT_ITT.2.2** **The TSF shall separate user data from TSF data when such data is transmitted between physically-separated parts of the TOE.**

**FPT_ITT.2.3** If the TSF provides multiple mechanisms to protect TSF data during transmission between physically-separated parts of the TOE, the TSF shall provide authorised administrators with the ability to select the method used.

**FPT_ITT.2.4** **The TSF shall restrict the ability to configure the separation mechanism to the authorised administrator.**

Dependencies : No dependencies.

### FPT_ITT.3 TSF Data Integrity Monitoring

Hierarchical to: FPT_ITT.1

**FPT_ITT.3.1** The TSF shall use [assignment: *specific mechanism*] to protect TSF data from [selection: *disclosure, modification, disclosure and modification*] when it is transmitted between physically-separated parts of the TOE.

**FPT_ITT.3.2** If the TSF provides multiple mechanisms to protect TSF data during transmission between physically-separated parts of the TOE, the TSF shall provide authorised administrators with the ability to select the method used.

**FPT_ITT.3.3** **The TSF shall be able to detect [selection: *modification of data, substitution of data, re-ordering of data, deletion of data,* [assignment: *other integrity errors*]] for TSF data transmitted between physically-separated parts of the TOE.**

**FPT_ITT.3.4** **Upon detection of a data integrity error, the TSF shall [assignment: *specify the action to be taken*].**

Dependencies : No dependencies.

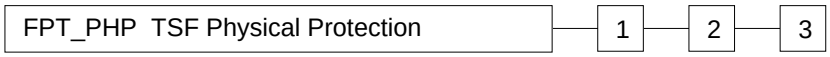
# FPT_PHP TSF Physical Protection

## Family behaviour

328 TSF physical protection components refer to restrictions on unauthorised physical access to the TSF, and to the deterrence of, and resistance to, unauthorised physical use, modification, or substitution of the TSF.

329 The requirements of components in this family ensure that the TSF is protected from physical tampering and interference. Satisfying the requirements of these components results in the TSF being packaged and used in such a manner that physical tampering is detectable, or resistance to physical tampering is measurable based on defined work factors. Without these components, the protection functions of a TSF lose their effectiveness in environments where physical damage cannot be prevented. This family also provides requirements regarding how the TSF shall respond to physical tampering attempts.

## Component levelling




330 FPT_PHP.1 Passive Detection of Physical Attack, provides for features that indicate when a TSF device or element is subject to tampering. However, notification of a tampering attack is not automatic; an administrator must invoke a security administrative function or perform manual inspection to determining if tampering has occurred.

331 FPT_PHP.2 Notification of Physical Attack, provides for automatic notification of tampering attacks for an identified subset of physical penetrations.

332 FPT_PHP.3 Resistance to Physical Attack, provides for features that prevent or resist physical tampering with TSF devices and elements.

## Audit :

333 Although there is not an explicit requirement to audit when a physical attack is detected, or an administrator is notified of an attack, this is solely because there is the potential that the detection and alarm mechanisms may be implemented completely in hardware, below the level of interaction with an audit subsystem (for example, a hardware-based detection system based on breaking a circuit and lighting an LED if the circuit is broken when a button is pressed by the administrator). Nevertheless, a PP/ST author may determine that for a particular anticipated threat environment there is a need to audit physical attacks. If this is the case, the PP/ST author should include appropriate requirements in the list of audit events. Note that inclusion of these requirements may have implications on the hardware design and its interface to the software.

### FPT_PHP.1 Passive Detection of Physical Attack

Hierarchical to: no other components.

**FPT_PHP.1.1** **The TOE shall include features that provide unambiguous detection of physical tampering with the TSF's physical devices and elements.**

**FPT_PHP.1.2** **The TSF shall provide the authorised administrator with the capability to determine whether physical tampering with the TSF's devices and elements has occurred.**

Dependencies : **AGD_ADM.1 Administrator guidance**
**FPT_TSA.1 Basic Security Administration**

### FPT_PHP.2 Notification of Physical Attack

Hierarchical to: FPT_PHP.1

**FPT_PHP.2.1** The TOE shall include features that provide unambiguous detection of physical tampering with the TSF's physical devices and elements.

**FPT_PHP.2.2** The TSF shall provide the authorised administrator with the capability to determine whether physical tampering with the TSF's devices and elements has occurred.

**FPT_PHP.2.3** **For [assignment: *list of devices/elements for which active detection is required*], the TSF shall monitor the devices and elements and notify [assignment: *a designated user or role*] when physical tampering with the TSF's devices and elements has occurred.**

Dependencies : AGD_ADM.1 Administrator guidance
FPT_TSA.1 Basic Security Administration

### FPT_PHP.3 Resistance to Physical Attack

Hierarchical to: FPT_PHP.2

**FPT_PHP.3.1** The TOE shall include features that provide unambiguous detection of physical tampering with the TSF's physical devices and elements.

**FPT_PHP.3.2** The TSF shall provide the authorised administrator with the capability to determine whether physical tampering with the TSF's devices and elements has occurred.

**FPT_PHP.3.3** For [assignment: *list of devices/elements for which active detection is required*], the TSF shall monitor the devices and elements and notify [assignment: *a designated user or role*] when physical tampering with the TSF's devices and elements has occurred.

**FPT_PHP.3.4** **For the following subset of the TSF's devices and elements, the TOE shall include features that resist identified physical tampering attacks to the TSF's devices and elements:**

**a)** **[assignment: *list of <devices/elements, physical tampering attack scenarios, work factors> for which resistance to attack is required*]**

**FPT_PHP.3.5** **For the following identified attack scenarios against the following subset of the TSF's device and elements, the TOE shall include features that automatically respond to the attack in such a way as to ensure that the TSP is not violated.**

**a)** **[assignment: *list of <devices/elements, physical tampering attack scenarios> for which automatic response to attack is required*]**

Dependencies : AGD_ADM.1 Administrator guidance

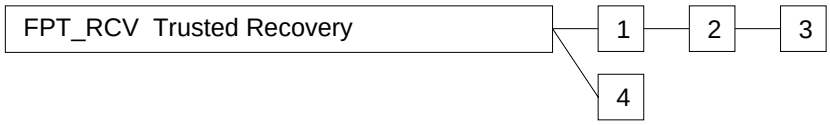
FPT_TSA.1 Basic Security Administration

# FPT_RCV Trusted Recovery

Family behaviour

334 The requirements of this family ensure that the TSF can determine that the TOE is started up without protection compromise and can recover without protection compromise after discontinuity of operations. Satisfying the requirements of this family establishes that the initial and recovered states of the TSF satisfy the requirements. This family is important because the start-up state of the TSF determines the protection of subsequent states.

Component levelling




335 FPT_RCV.1 Manual Recovery, allows a TOE to provide only mechanisms that involve human intervention to return to a secure state.

336 FPT_RCV.2 Automated Recovery, provides, for at least one type of service discontinuity, recovery to a secure state without human intervention; recovery for other discontinuities may require human intervention.

337 FPT_RCV.3 Automated Recovery without Undue Loss, also provides for automated recovery, but strengthens the requirements by disallowing undue loss of protected objects.

338 FPT_RCV.4 Function Recovery, provides for recovery at the level of particular SFs, ensuring either successful completion or rollback of TSF data to the state before the function was invoked.

## FPT_RCV.1 Manual Recovery

Hierarchical to: no other components.

FPT_RCV.1.1 **After a failure or service discontinuity, the TSF shall enter a maintenance mode where the ability to return the TOE to a secure state is provided.**

FPT_RCV.1.2 **The TSF shall provide the authorised administrator with the capability to restore the TSF data to a consistent and secure state.**

Dependencies : **FPT_TSA.1 Basic Security Administration**
**FPT_TST.1 On-Demand TSF Testing**
**AGD_ADM.1 Administrator guidance**
**ADV_FSP.2 Informal security policy model**

### FPT_RCV.2 Automated Recovery

Hierarchical to: FPT_RCV.1

**FPT_RCV.2.1** **When automated recovery from** a failure or service discontinuity **is not possible**, the TSF shall enter a maintenance mode where the ability to return the TOE to a secure state is provided.

**FPT_RCV.2.2** The TSF shall provide the authorised administrator with the capability to restore the TSF data to a consistent and secure state.

**FPT_RCV.2.3** **For [assignment: *list of failures/service discontinuties*], the TSF shall ensure the return of the TOE to a secure state using automated procedures.**

Dependencies : FPT_TSA.1 Basic Security Administration
FPT_TST.1 On-Demand TSF Testing
AGD_ADM.1 Administrator guidance
ADV_FSP.2 Informal security policy model

### FPT_RCV.3 Automated Recovery without Undue Loss

Hierarchical to: FPT_RCV.2

**FPT_RCV.3.1** When automated recovery from a failure or service discontinuity is not possible, the TSF shall enter a maintenance mode where the ability to return the TOE to a secure state is provided.

**FPT_RCV.3.2** The TSF shall provide the authorised administrator with the capability to restore the TSF data to a consistent and secure state.

**FPT_RCV.3.3** For [assignment: *list of failures/service discontinuties*], the TSF shall ensure the return of the TOE to a secure state using automated procedures.

**FPT_RCV.3.4** **The functions provided by the TSF to recover from failure or service discontinuity shall ensure that the secure initial state is restored without exceeding [assignment: *quantification*] for loss of TSF data or objects within the TSC.**

**FPT_RCV.3.5** **The TSF shall provide the authorised administrator with the capability to determine the objects that were or were not capable of being recovered.**

Dependencies : FPT_TSA TOE Security Administration
FPT_TST.1 On-Demand TSF Testing
AGD_ADM.1 Administrator guidance
ADV_FSP.2 Informal security policy model

## FPT_RCV.4 Function Recovery

Hierarchical to: no other components.

FPT_RCV.4.1 **The TSF shall ensure that [assignment: *list of SFs and failure scenarios*] have the property that the SF either completes successfully, or for the indicated failure scenarios, recovers to a state that is the state immediately before the invocation of the SF.**

Dependencies : **FPT_TSA.1 Basic Security Administration**
**FPT_TST.1 On-Demand TSF Testing**
**AGD_ADM.1 Administrator guidance**
**ADV_FSP.2 Informal security policy model**

# FPT_REV Revocation

Family behaviour

339 This family addresses revocation of security attributes for a variety of entities within a TOE.

Component levelling


340 FPT_REV.1 Basic Revocation, provides for revocation of security attributes to be enforced at some point in time.

341 FPT_REV.2 Immediate Revocation, provides for immediate revocation of security attributes.

## FPT_REV.1 Basic Revocation

Hierarchical to: no other components.

**FPT_REV.1.1** **The TSF shall provide a capability for revocation of security attributes associated with the [selection: *users, subjects, objects,* [assignment: *list of additional resources]*] within the TSC.**

**FPT_REV.1.2** **The TSF shall enforce revocation [assignment: *specification of revocation rules*].**

Dependencies : No dependencies.

## FPT_REV.2 Immediate Revocation

Hierarchical to: FPT_REV.1

**FPT_REV.2.1** The TSF shall provide a capability for revocation of security attributes associated with the [selection: *users*, *subjects*, *objects*, [assignment: *list of additional resources]*] within the TSC.

**FPT_REV.2.2** The TSF shall **immediately** enforce revocation **of security attributes.**

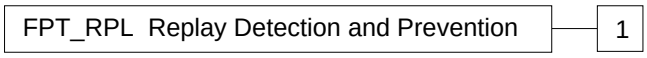
Dependencies : No dependencies.

# FPT_RPL Replay Detection and Prevention

Family behaviour

342 This family addresses detection of replay for various types of entities. In the case where all forms of replay may be detected, this effectively prevents replay.

Component levelling




343 The family consists of only one component, FPT_RPL.1 Replay Detection and Prevention, which requires that the TSF shall be able to detect the replay of identified entities (e.g., messages, service requests, service responses).

Audit :

344 The following actions should be audited if FAU Security Audit is included in the PP/ST:

a) Basic: Detected replay attacks.

## FPT_RPL.1 Replay Detection and Prevention

Hierarchical to: no other components.

**FPT_RPL.1.1 The TSF shall detect replay for [assignment: *list of identified entities*].**

**FPT_RPL.1.2 The TSF shall be able to take [assignment: *list of specific actions*] to be taken when replay is detected.**

Dependencies : No dependencies.

# FPT_RVM Reference Mediation

Family behaviour

345 The components of this family address the "always invoked" aspect of a traditional reference monitor. The goal of these components is to ensure, with respect to a given SFP, that all actions requiring policy enforcement are validated by the TSF against the SFP. If the portion of the TSF that enforces the SFP also meets the requirements of appropriate components from FPT_SEP (Domain Separation) and ADV_INT (TSF internals), than that portion of the TSF provides a "reference monitor" for that SFP.

346 A TSF that implements a SFP provides effective protection against unauthorised operation if and only if all enforceable actions (e.g., accesses to objects) issued by untrusted subjects with respect to any or all of that SFP's subjects are validated by the TSF before succeeding. If the enforceable action is incorrectly enforced or bypassed, the overall enforcement of the SFP has been compromised. "Untrusted" subjects could then bypass the SFP in a variety of unauthorised ways (e.g., circumvent access checks for some subjects or objects, bypass checks for objects whose protection was assumed by applications, retain access rights beyond their intended lifetime, bypass auditing of audited actions, or bypass authentication). Note that the term "untrusted subject" refers to subjects untrusted with respect to the any or all of specific SFPs being enforced; a subject may be trusted with respect to one SFP and untrusted with respect to a different SFP.

Component levelling


347 This family consists of only one component, FPT_RVM.1 Non-Bypassability of the TSP, which requires non-bypassability for all SFPs in the TSP.

## FPT_RVM.1 Non-Bypassability of the TSP

Hierarchical to: no other components.

**FPT_RVM.1.1 The TSF shall ensure that TSP enforcement functions are invoked and succeed before any security-related operation is allowed to proceed.**

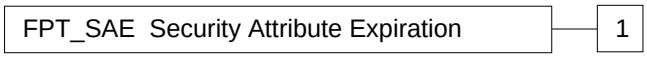
Dependencies : No dependencies.

# FPT_SAE Security Attribute Expiration

Family behaviour

348 This family addresses the capability to enforce time limits for the validity of security attributes.

Component levelling




349 This family consists of only one component, FPT_SAE.1 Time-Limited Authorisation, which requires the ability for the authorised administrator to specify an expiration time on specified security attributes.

Audit :

350 The following actions should be audited if FAU Security Audit is included in the PP/ST:

a) Basic: Specification of the expiration time for an attribute

## FPT_SAE.1 Time-Limited Authorisation

Hierarchical to: no other components.

**FPT_SAE.1.1** **The TSF shall provide a capability for the authorised administrator to specify an expiration time for [assignment: *list of security attributes for which expiration is to be supported*].**

**FPT_SAE.1.2** **For each of these security attributes, the TSF shall be able to [assignment: *list of actions to be taken for each security attribute*] after the expiration time for the indicated security attribute has passed.**

Dependencies : No dependencies.

# FPT_SEP Domain Separation

## Family behaviour

351 The components of this family ensure that at least one security domain is available for the TSF's own execution, and that the TSF is protected from external interference and tampering (e.g., by modification of TSF code or data structures) by untrusted subjects. Satisfying the requirements of this family makes the TSF self-protecting, meaning that an untrusted subject cannot modify or damage the TSF.

352 This family requires the following:

a) The resources of the TSF's security domain ("protected domain") and those of subjects and unconstrained entities external to the domain are separated such that the entities external to the protected domain cannot observe or modify TSF data or TSF code internal to the protected domain.

b) The transfers between domains are controlled such that arbitrary entry to, or return from, the protected domain is not possible.

c) The user or application parameters passed to the protected domain by addresses are validated with respect to the protected domain's address space, and those passed by value are validated with respect to the values expected by the protected domain.

d) The security domains of subjects are distinct except for controlled sharing via the TSF.

## Component levelling


353 FPT_SEP.1 TSF Domain Separation, provides a distinct protected domain for the TSF and provides separation between subjects within the TSC.

354 FPT_SEP.2 Reference Monitor for some SFPs, requires that the TSF be further subdivided, with distinct domain(s) for an identified set of SFPs that act as reference monitors for their policies, and a domain for the remainder of the TSF, as well as domains for the non-TSF portions of the TOE.

355 FPT_SEP.3 Complete Reference Monitor, requires that there be distinct domain(s) for TSP enforcement, a domain for the remainder of the TSF, as well as domains for the non-TSF portions of the TOE.

### FPT_SEP.1 TSF Domain Separation

Hierarchical to: no other components.

**FPT_SEP.1.1** **The TSF shall maintain a security domain for its own execution that protects it from interference and tampering by untrusted subjects.**

**FPT_SEP.1.2** **The TSF shall enforce separation between the security domains of subjects in the TSC.**

Dependencies : No dependencies.

### FPT_SEP.2 Reference Monitor for some SFPs

Hierarchical to: FPT_SEP.1

**FPT_SEP.2.1** The **unisolated portion of the** TSF shall maintain a security domain for its own execution that protects it from interference and tampering by untrusted subjects.

**FPT_SEP.2.2** The TSF shall enforce separation between the security domains of subjects in the TSC.

**FPT_SEP.2.3** **The TSF shall maintain [assignment: *list of access control and information flow SFPs*] in a security domain for their own execution that protects them from interference and tampering by the remainder of the TSF and by subjects untrusted with respect to those SFPs.**

Dependencies : No dependencies.

### FPT_SEP.3 Complete Reference Monitor

Hierarchical to: FPT_SEP.2

**FPT_SEP.3.1** The **non-TSP enforcing** portion of TSF shall maintain a security domain for its own execution that protects it from interference and tampering by untrusted subjects.

**FPT_SEP.3.2** The TSF shall enforce separation between the security domains of subjects in the TSC.

**FPT_SEP.3.3** The TSF shall maintain **the portion of the TSF that enforces the access control and information flow SFPs** in a security domain for **its** own execution that protects **it** from interference and tampering by the remainder of the TSF and by subjects untrusted with respect to **the TSP**.

Dependencies : No dependencies.

# FPT_SSP State Synchrony Protocol

Family behaviour

355 A myriad of actions in distributed systems gain complexity over their mainframe equivalents for reasons such as message time delay and state synchrony revocation, permission, encryption key invocation, audit, and database update. In most cases synchronisation of state between distributed functions involves an exchange protocol, not a simple action. When malice exists in the distributed environment of these protocols, more complex defensive protocols are required.

355 FPT_SSP establishes the requirement for certain critical security functions of the TSF to use this trusted protocol. FPT_SSP ensures that two distributed parts of the TOE (e.g., hosts) have synchronised their states after a security-relevant action.

Component levelling


356 FPT_SSP.1 Simple Trusted Acknowledgement requires only a simple acknowledgment by the data recipient.

357 FPT_SSP.2 Mutual Trusted Acknowledgement requires mutual acknowledgment of the data exchange.

## FPT_SSP.1 Simple Trusted Acknowledgement

Hierarchical to: no other components.

**FPT_SSP.1.1** **The TSF shall be able to acknowledge, when requested by another part of the TSF, the receipt of an unmodified TSF data transmission.**

Dependencies : **FPT_ITI.1 Inter-TSF Detection of Modification**
**FTP_ITC.1 Inter-TSF Trusted Channel**

## FPT_SSP.2 Mutual Trusted Acknowledgement

Hierarchical to: FPT_SSP.1

**FPT_SSP.2.1** The TSF shall be able to acknowledge, when requested by another part of the TSF, the receipt of an unmodified TSF data transmission.

**FPT_SSP.2.2** **The TSF shall provide the ability for the receiving TSF to request an acknowledgment that the acknowledgement of message receipt was received by the sending TSF.**

Dependencies : FPT_ITI.1 Inter-TSF Detection of Modification
FTP_ITC.1 Inter-TSF Trusted Channel

# FPT_STM Time Stamps

Family behaviour

357 This family addresses requirements for a trusted time stamp function within a TOE.

Component levelling


357 This family consists of only one component, FPT_STM.1 Trusted Time Stamps, which requires that the TSF provide trusted time stamps for TSF functions.

## FPT_STM.1 Trusted Time Stamps

Hierarchical to: no other components.

**FPT_STM.1.1 The TSF shall be able to provide reliable time stamps for TSF functions.**

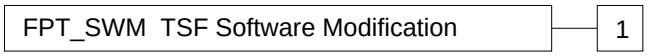
Dependencies : No dependencies.

# FPT_SWM TSF Software Modification

Family behaviour

358 The requirements of this family are needed to detect the corruption of TSF code by various failures that do not necessarily stop the TOE's operation (which would be handled by other families). These checks must be performed because these failures may not necessarily be prevented. Such failures can occur either because of unforeseen failure modes or associated oversights in the design of hardware, firmware, or software, or because of malicious corruption of the TSF due to inadequate logical and/or physical protection.

Component levelling




359 This family consists of only one component, FPT_SWM.1 Protection of Executables, which requires that the TSF be able to verify the integrity of the TSF executable code.

## FPT_SWM.1 Protection of Executables

Hierarchical to: no other components.

**FPT_SWM.1.1 The TSF shall provide the authorised administrator with the capability to verify the integrity of stored TSF executable code.**

Dependencies : No dependencies.

# FPT_TDC Inter-TSF TSF Data Consistency

Family behaviour

359 In a distributed or composite system environment, a TOE may need to exchange TOE data (e.g., the SFP-attributes associated with data, audit information, identification information) with the TSF of another distinct TOE. This family defines the requirements for sharing and consistent interpretation of these attributes between the TSFs of different TOEs.

Component levelling


360 FPT_TDC.1 Inter-TSF Basic TSF Data Consistency requires that the TSF provide mechanisms to ensure consistency of attributes between TSFs.

Audit :

361 The following actions should be audited if FAU Security Audit is included in the PP/ST:

a) Minimal: Successful use of TSF data consistency mechanisms.

b) Basic: Any use of the TSF data consistency mechanisms.

c) Basic: Identification of which TSF data have been interpreted.

d) Basic: Detection of modified TSF data.

e) Detailed: Recovery of original TSF data sent.

f) Detailed: Capture of the actual values of each TSF data item.

## FPT_TDC.1 Inter-TSF Basic TSF Data Consistency

Hierarchical to: no other components.

**FPT_TDC.1.1 The TSF shall enforce the consistent interpretation of [assignment: *list of TSF data types*] during inter-TSF transfers.**

**FPT_TDC.1.2 The TSF shall use [assignment: *list of interpretation rules to be applied by the TSF*] when interpreting the TSF data during Inter-TSF transfers.**

Dependencies : No dependencies.

# FPT_TRC Internal TOE TSF Data Replication Consistency

Family behaviour

362 The requirements of this family are needed to ensure the consistency of TSF data when such data is replicated internal to the TOE. Such data may become inconsistent if the internal channel between parts of the TOE becomes inoperative. If the TOE is internally structured as a network of parts of the TOE, this can occur when parts become disabled, network connections are broken, and so on.

Component levelling


363 This family consists of only one component, FPT_TRC.1 Internal TOE Data Consistency, which requires that the TSF ensure the consistency of TSF data that is replicated in multiple locations.

## FPT_TRC.1 Internal TOE Data Consistency

Hierarchical to: no other components.

**FPT_TRC.1.1 The TSF shall ensure that TSF data is consistent when replicated between parts of the TOE.**

**FPT_TRC.1.2 When parts of the TOE containing replicated TSF data are disconnected, the TSF shall ensure the consistency of the replicated TSF data upon reconnection before processing any requests for [assignment: *list of SFs dependent on TSF data replication consistency*].**

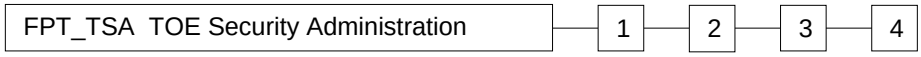
Dependencies : **FPT_ITT.1 Basic Internal TSF Data Transfer Protection**

## FPT_TSA TOE Security Administration

### Family behaviour

364 The TSF includes security administration families to allow authorised administrators to control the secure operation of the TOE, and to restrict the accessibility of security management functions to authorised users. At higher levels of this family, the administrative function is subdivided into distinct roles.

### Component levelling




365 FPT_TSA.1 Basic Security Administration, requires that sufficient functions be available through the TSF to securely install, configure, and manage to the TOE; and that these functions be identified and accessible to authorised users only.

366 FPT_TSA.2 Separate Security Administrative Role, elaborates on this by establishing a distinct administrative role to which these functions are restricted; explicit action is required to assume this role.

367 FPT_TSA.3 Multiple Security Administrative Roles, provides further restrictions on the administrator by dividing the security-relevant TSF functions into multiple administrative roles (for example, administrator and operator, auditor, account administrator).

368 FPT_TSA.4 Well-Defined Administrative Roles, further restricts each of the administrative roles by limiting the functions available in an administrative role to the minimal set required to act in that role.

### Audit :

369 The following actions of this component should be audited if FAU_GEN Security Audit Data Generation is included in the PP/ST:

a) Minimal: Use of a security-relevant administrative function.

### Audit : for FPT_TSA.2

370 In addition to the audit required for all components in this family, the following actions of this component should also be audited if FAU_GEN Security Audit Data Generation is included in the PP/ST:

a) Minimal: Explicit requests to assume the security administrative role

b) Basic: The allocation of a function to a security administrative role.

Audit : for FPT_TSA.3

371 In addition to the audit required for all components in this family, the following actions of this component should also be audited if FAU_GEN Security Audit Data Generation is included in the PP/ST:

a) Minimal: Explicit requests to assume a security administrative role

b) Basic: The addition or deletion of a user to/from a security administrative role.

c) Basic: The association of a security-relevant administrative function with a specific security administrative role.

Audit : for FPT_TSA.4

372 In addition to the audit required for all components in this family, the following actions of this component should also be audited if FAU_GEN Security Audit Data Generation is included in the PP/ST:

a) Minimal: Explicit requests to assume a security administrative role

b) Basic: The addition or deletion of a user to/from a security administrative role.

c) Basic: The association of a security-relevant administrative function with a specific security administrative role.

### FPT_TSA.1 Basic Security Administration

Hierarchical to: no other components.

**FPT_TSA.1.1** **The TSF shall distinguish security-relevant administrative functions from other functions.**

**FPT_TSA.1.2** **The TSF's set of security-relevant administrative functions shall include all functions necessary to install, configure, and manage the TSF; minimally, this set shall include [assignment: *list of administrative services to be minimally supplied*].**

**FPT_TSA.1.3** **The TSF shall restrict the ability to perform security-relevant administrative functions to specifically authorised users.**

**FPT_TSA.1.4** **The TSF shall be capable of distinguishing the set of users authorised for administrative functions from the set of all users of the TOE.**

Dependencies : **FIA_UID.1 Basic User Identification**
**FIA_ATD.1 User Attribute Definition**
**FIA_ATA.1 User Attribute Initialisation**
**AGD_ADM.1 Administrator guidance**

### FPT_TSA.2 Separate Security Administrative Role

Hierarchical to: FPT_TSA.1

**FPT_TSA.2.1** The TSF shall distinguish security-relevant administrative functions from other functions.

**FPT_TSA.2.2** The TSF's set of security-relevant administrative functions shall include all functions necessary to install, configure, and manage the TSF; minimally, this set shall include [assignment: *list of administrative services to be minimally supplied*].

**FPT_TSA.2.3** The TSF shall restrict the ability to perform security-relevant administrative functions to **a security administrative role that has a specific set of authorised functions and responsibilities**.

**FPT_TSA.2.4** The TSF shall be capable of distinguishing the set of users authorised for administrative functions from the set of all users of the TOE.

**FPT_TSA.2.5** **The TSF shall allow only specifically authorised users to assume the security administrative role.**

**FPT_TSA.2.6** **The TSF shall require an explicit request to be made in order for an authorised user to assume the security administrative role.**

Dependencies : FIA_UID.1 Basic User Identification
FIA_ATD.1 User Attribute Definition
FIA_ATA.1 User Attribute Initialisation
AGD_ADM.1 Administrator guidance

### FPT_TSA.3 Multiple Security Administrative Roles

Hierarchical to: FPT_TSA.2

**FPT_TSA.3.1** The TSF shall distinguish security-relevant administrative functions from other functions.

**FPT_TSA.3.2** The TSF's set of security-relevant administrative functions shall include all functions necessary to install, configure, and manage the TSF; minimally, this set shall include **assignment/deletion of authorised users from security administrative roles, association of security-relevant administrative commands with security administrative roles, and** [assignment: *list of administrative services to be minimally supplied*].

**FPT_TSA.3.3** The TSF shall restrict the ability to perform **a** security-relevant administrative function to **the** security administrative role**(s) authorised to use that function**.

**FPT_TSA.3.4** The TSF shall be capable of distinguishing the set of users authorised for administrative functions from the set of all users of the TOE.

**FPT_TSA.3.5** The TSF shall allow only specifically authorised users to assume **only those** security administrative role**s for which they have been authorised.**

**FPT_TSA.3.6** The TSF shall require an explicit request **to assume a specific security administrative role** to be made in order for an authorised user to assume **that** security administrative role.

**FPT_TSA.3.7** **The TSF shall define a set of security administrative roles that minimally includes [assignment: *set of defined roles to be minimally supported*].**

**FPT_TSA.3.8** **The TSF shall associate each security-relevant administrative function with at least one security administrative role.**

Dependencies : FIA_UID.1 Basic User Identification
FIA_ATD.1 User Attribute Definition
FIA_ATA.1 User Attribute Initialisation
AGD_ADM.1 Administrator guidance

### FPT_TSA.4 Well-Defined Administrative Roles

Hierarchical to: FPT_TSA.3

**FPT_TSA.4.1** The TSF shall distinguish security-relevant administrative functions from other functions.

**FPT_TSA.4.2** The TSF's set of security-relevant administrative functions shall include all functions necessary to install, configure, and manage the TSF; minimally, this set shall include assignment/deletion of authorised users from security administrative roles, association of security-relevant administrative commands with security administrative roles, and [assignment: *list of additional administrative services to be minimally supplied*].

**FPT_TSA.4.3** The TSF shall restrict the ability to perform a security-relevant administrative function to the security administrative role(s) authorised to use that function.

**FPT_TSA.4.4** The TSF shall be capable of distinguishing the set of users authorised for administrative functions from the set of all users of the TOE.

**FPT_TSA.4.5** The TSF shall allow only specifically authorised users to assume only those security administrative roles for which they have been authorised.

**FPT_TSA.4.6** The TSF shall require an explicit request to assume a specific security administrative role to be made in order for an authorised user to assume that security administrative role.

**FPT_TSA.4.7** The TSF shall define a set of security administrative roles that minimally includes [assignment: *set of defined roles to be minimally supported*].

**FPT_TSA.4.8** The TSF shall associate each security-relevant administrative function with **only** one security administrative role.

**FPT_TSA.4.9** **The TSF shall assign to each security administrative user role only those functions strictly required to perform that role effectively.**

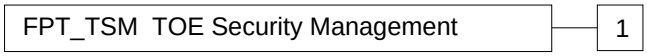
Dependencies : FIA_UID.1 Basic User Identification
FIA_ATD.1 User Attribute Definition
FIA_ATA.1 User Attribute Initialisation
AGD_ADM.1 Administrator guidance

# FPT_TSM TOE Security Management

Family behaviour

373 The TSF of a TOE should provide security management functions to enable authorised administrators to set up and control the secure operation of the product. This family provides requirements for these administrative functions.

Component levelling




374 This family consists of only one component, FPT_TSM.1 Management Functions, which requires that the TSF allow authorised administrators to set and update TSF configuration parameters.

Audit :

375 The following actions of this component should be audited if FAU_GEN Security Audit Data Generation is included in the PP/ST:

a) Basic: Successful and unsuccessful attempts to modify (set and update) TSF configuration parameters.

## FPT_TSM.1 Management Functions

Hierarchical to: no other components.

**FPT_TSM.1.1 The TSF shall provide the authorised administrator with the ability to set and update [assignment: *list of TSF configuration parameters*].**

**FPT_TSM.1.2 The TSF shall provide the authorised administrator with the ability to perform [assignment: *list of desired administrative functions*].**

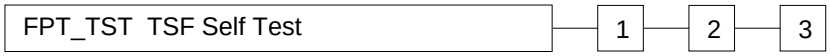
Dependencies : **FPT_TSA.1 Basic Security Administration**

# FPT_TST TSF Self Test

Family behaviour

376 The family defines the requirements for the definition of self-testing of the TSF with respect to some expected correct operation. Examples are calls to enforcement functions, and sample arithmetical operations on critical parts of the TOE. These tests can be carried out either in some maintenance state, at start-up, on-line, or continuously. The actions to be taken by the TOE as the result of self testing are defined in other families.

Component levelling




377 FPT_TST.1 On-Demand TSF Testing, provides the ability to test the TSF's correct operation. These tests may be performed during normal operation, in a maintenance mode, or off-line.

378 FPT_TST.2 TSF Testing During Start-Up, provides for both human-invoked periodic tests and TSF-invoked testing during start-up.

379 FPT_TST.3 TSF Testing During Normal Operation, provides for periodic TSF-invoked testing of the TSF's correct operation during normal operation, as well as administrator-invoked tests and testing during start-up.

Audit :

380 No audit in addition to that required for security administrative functions, if the periodic tests are invoked by the administrator during normal operation.

## FPT_TST.1 On-Demand TSF Testing

Hierarchical to: no other components.

**FPT_TST.1.1 The TSF shall provide authorised administrators with the capability to demonstrate the correct operation of the TSF.**

**FPT_TST.1.2 The TSF shall provide authorised administrators with the capability to verify the integrity of TSF data.**

Dependencies : **FPT_AMT.1 Abstract Machine Testing**

### FPT_TST.2 TSF Testing During Start-Up

Hierarchical to: FPT_TST.1

**FPT_TST.2.1** The TSF shall provide authorised administrators with the capability to demonstrate the correct operation of the TSF.

**FPT_TST.2.2** The TSF shall provide authorised administrators with the capability to verify the integrity of TSF data.

**FPT_TST.2.3** **The TSF shall exercise a suite of self tests during initial start-up in order to demonstrate the correct operation of the TSF.**

Dependencies : **FPT_AMT.2 Abstract Machine Testing During Start-Up**

### FPT_TST.3 TSF Testing During Normal Operation

Hierarchical to: FPT_TST.2

**FPT_TST.3.1** The TSF shall provide authorised administrators with the capability to demonstrate the correct operation of the TSF.

**FPT_TST.3.2** The TSF shall provide authorised administrators with the capability to verify the integrity of TSF data.

**FPT_TST.3.3** The TSF shall exercise a suite of self tests during initial start-up **and periodically during normal operation** in order to demonstrate the correct operation of the TSF.

Dependencies : **FPT_AMT.3 Abstract Machine Testing During Normal Operation**

# FPT_TSU TOE Administrative Safe Use

Family behaviour

381 The elements of this family address general characteristics of TSF administrative interfaces that reduce the likelihood that an unskilled authorised administrator will use a TSF interface in an insecure manner. To some extent, these components address ease of use of the administrative function; however, ease of use is a subjective measure that cannot be precisely measured or evaluated.

Component levelling


382 FPT_TSU.1 Enforcement of Administrative Guidance, simply requires that the TSF enforce any bounds restrictions described in the Administrative Guidance.

383 FPT_TSU.2 Safe Administrative Defaults, additionally requires that the TSF provide specific interfaces and options with fail-safe defaults.

384 FPT_TSU.3 Administrator Defined Defaults, requires that the TSF provide the authorised administrator with the ability to modify the values of these defaults.

## FPT_TSU.1 Enforcement of Administrative Guidance

Hierarchical to: no other components.

**FPT_TSU.1.1** **The TSF shall enforce checks for valid input values for security-relevant administrative functions as described in the Administrative Guidance.**

Dependencies : **FPT_TSA.1 Basic Security Administration**
**AGD_ADM.1 Administrator guidance**

## FPT_TSU.2 Safe Administrative Defaults

Hierarchical to: FPT_TSU.1

**FPT_TSU.2.1** The TSF shall enforce checks for valid input values for security-relevant administrative functions as described in the Administrative Guidance.

**FPT_TSU.2.2** **The TSF shall provide safe default values for [assignment: *list of security attributes*].**

Dependencies : FPT_TSA.1 Basic Security Administration
AGD_ADM.1 Administrator guidance

### FPT_TSU.3 Administrator Defined Defaults

Hierarchical to: FPT_TSU.2

**FPT_TSU.3.1** The TSF shall enforce checks for valid input values for security-relevant administrative functions as described in the Administrative Guidance.

**FPT_TSU.3.2** The TSF shall provide safe default values for [assignment: *list of security attributes*].

**FPT_TSU.3.3** **The TSF shall provide the authorised administrator with the ability to specify the default values for [assignment: *list of security attributes*].**

Dependencies : FPT_TSA.1 Basic Security Administration
AGD_ADM.1 Administrator guidance

# Class FRU

# Resource Utilisation

385 This class provides three families which support the availability of required resources such as processing capability and/or storage capacity when needed. The family Fault Tolerance provides protection against unavailability of capabilities caused by failure of the TOE. The family Priority of Service ensures that the resources will be allocated to the more important or time-critical tasks and cannot be monopolised by lower priority tasks. The family Resource Allocation provides limits on the use of available resources, therefore preventing users from monopolising the resources.

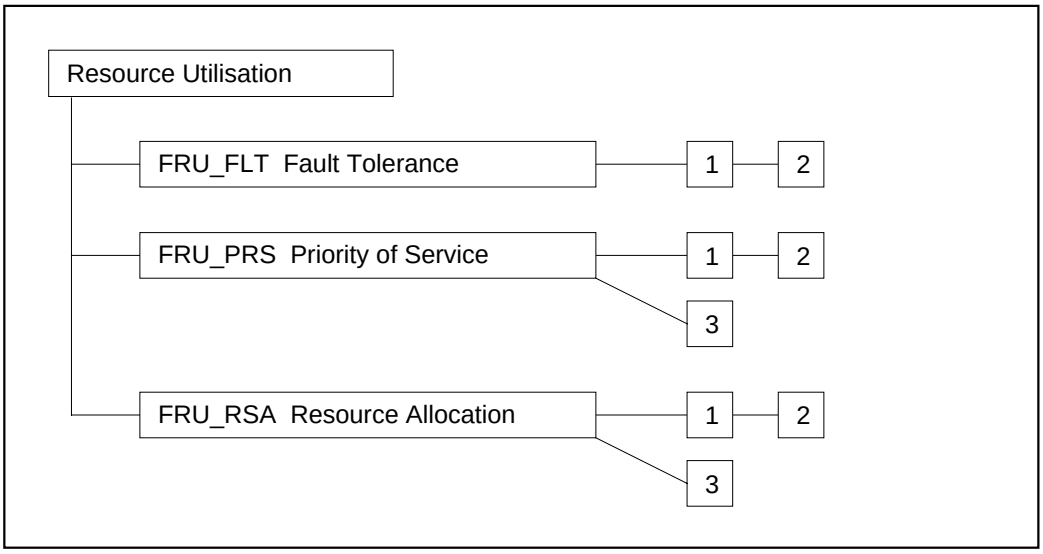


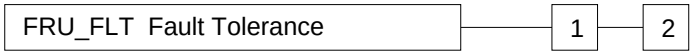

**Figure 2.15 - Resource Utilisation class decomposition**

## FRU_FLT Fault Tolerance

Family behaviour

386 The requirements of this family ensure that the TOE will maintain correct operation even in the event of failures.

Component levelling




387 FRU_FLT.1 Degraded Fault Tolerance requires the TOE to continue correct operation of identified capabilities in the event of identified failures.

388 FRU_FLT.2 Limited Fault Tolerance requires the TOE to continue correct operation of all capabilities in the event of identified failures.

Audit : for FRU_FLT.1 and FRU_FLT.2

389 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: Any failure detected by the TSF.

Audit : for FRU_FLT.1

390 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Basic: All functions being discontinued due to a failure.

### FRU_FLT.1 Degraded Fault Tolerance

Hierarchical to: no other components.

FRU_FLT.1.1 **The TSF shall continue its operation of [assignment: *list of TOE capabilities*] that will be maintained when [assignment: *list of type of failures*] occur.**

Dependencies : **FPT_FLS.1 Failure with Preservation of Secure State**
**ADV_FSP.1 TOE and security policy**

### FRU_FLT.2 Limited Fault Tolerance

Hierarchical to: FRU_FLT.1

**FRU_FLT.2.1** The TSF shall continue its operation of **all its capabilities** when [assignment: *list of type of failures*] occur.

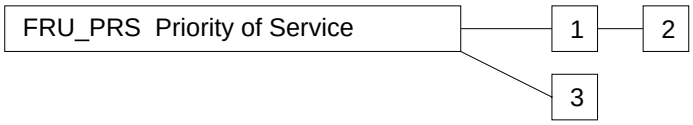
Dependencies : FPT_FLS.1 Failure with Preservation of Secure State
ADV_FSP.1 TOE and security policy

# FRU_PRS Priority of Service

## Family behaviour

391 The requirements of this family allow the TSF to control the use of resources within the TSC by subjects such that high priority activities within the TSC will always be accomplished without undue interference or delay caused by low priority activities.

## Component levelling




392 FRU_PRS.1 Limited Priority of Service provides priorities for a subject's use of a subset of the resources within the TSC.

393 FRU_PRS.2 Full Priority of Service provides priorities for a subject's use of all of the resources within the TSC.

394 FRU_PRS.3 Priority of Service Management provides the authorised administrator with the capability to set the priority of service of a subject.

Audit : for FRU_PRS.1, and FRU_PRS.2

395 The following actions shall be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST:

a) Minimal: Rejection of operation based on the use of priority within an allocation.

b) Basic: All attempted uses of the allocation function which involves the priority of the service functions.

Audit : for FRU_PRS.3

396 The following actions shall be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST:

a) Minimal: Use of the Priority of Service Management capability and the authorised administrator that invoked this service.

## FRU_PRS.1 Limited Priority of Service

Hierarchical to: no other components.

**FRU_PRS.1.1 The TSF shall assign a priority to each subject in the TSF.**

**FRU_PRS.1.2** **The TSF shall ensure that each access to [assignment: *controlled resources*] shall be subjected to the Priority of Service mechanism.**

Dependencies : No dependencies.

### FRU_PRS.2 Full Priority of Service

Hierarchical to: FRU_PRS.1

**FRU_PRS.2.1** The TSF shall assign a priority to each subject in the TSF.

**FRU_PRS.2.2** The TSF shall ensure that each access to **all shareable resources** shall be subjected to the Priority of Service mechanism.

Dependencies : No dependencies.

### FRU_PRS.3 Priority of Service Management

Hierarchical to: no other components.

**FRU_PRS.3.1** **The TSF shall allow the authorised administrator to assign a priority to each subject in the TSF.**

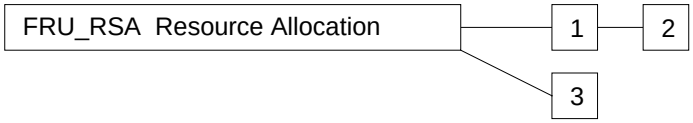
Dependencies : **FRU_PRS.1 Limited Priority of Service**
**FPT_TSA.1 Basic Security Administration**
**FIA_UID.1 Basic User Identification**

# FRU_RSA Resource Allocation

Family behaviour

397 The requirements of this family allow the TSF to control the use of resources by users such that denial of service will not occur because of unauthorised monopolisation of resources by users.

Component levelling




398 FRU_RSA.1 Maximum Quotas provides requirements for quota mechanisms that ensure that users will not monopolize a controlled resource.

399 FRU_RSA.2 Minimum and Maximum Quotas provides requirements for quota mechanisms that ensure that users will always have at least a minimum of a specified resource and that they will not be able to monopolise a controlled resource.

400 FRU_RSA.3 Quota Management provides requirements for the authorised administrator to define the quotes for an individual user or groups of users.

Audit :

401 The following actions shall be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST:

a) Minimal: Rejection of allocation operation due to resource limits.

b) Basic: All attempted uses of the resource allocation functions for resources that are under control of the TSF.

## FRU_RSA.1 Maximum Quotas

Hierarchical to: no other components.

FRU_RSA.1.1 **The TSF shall enforce quotas limiting the maximum quantity of [assignment: *controlled resources*] that [selection: *individual user, defined group of users*] can use [selection: *simultaneously, over a specified period of time*].**

Dependencies : **FIA_UID.1 Basic User Identification**

### FRU_RSA.2 Minimum and Maximum Quotas

Hierarchical to: FRU_RSA.1

**FRU_RSA.2.1** The TSF shall enforce quotas limiting the maximum quantity of [assignment: *controlled resources*] that [selection: *individual user, defined group of users*] can use [selection: *simultaneously, over a specified period of time*].

**FRU_RSA.2.2** **The TSF shall ensure the provision of minimum quantity of each [assignment: *controlled resource*] that is available for [selection: *an individual user, defined group of users*] to use [selection: *simultaneously, over a specified period of time*]**

Dependencies : FIA_UID.1 Basic User Identification

### FRU_RSA.3 Quota Management

Hierarchical to: no other components.

**FRU_RSA.3.1** **For each controlled resource, the TSF shall allow the authorised administrator to set the resource allocation limits to the granularity of [selection: *individual user, defined group of users*].**

Dependencies : **FRU_RSA.1 Maximum Quotas**

**FPT_TSA.1 Basic Security Administration**

# Class FTA

# TOE Access

402 This family specifies functional requirements, over and above identification and authentication requirements, for controlling the establishment of a user's session.

403 Figure 2.16 shows the decomposition of this class into its constituent components.

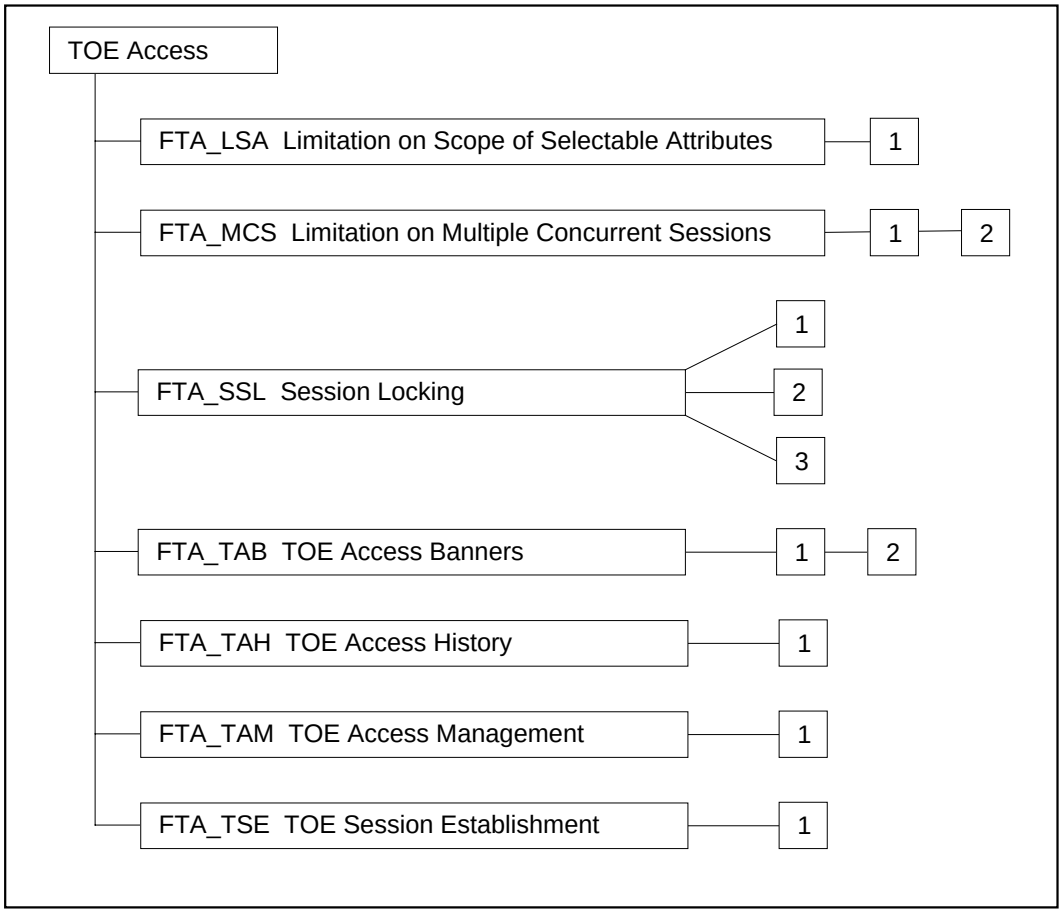


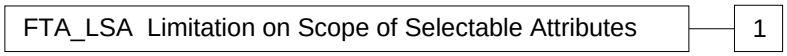

**Figure 2.16 - TOE Access class decomposition**

# FTA_LSA Limitation on Scope of Selectable Attributes

Family behaviour

404 This family defines requirements to limit the scope of attributes that a user may select for a session, based on environmental conditions.

Component levelling




405 There is only one component in this family.

Audit :

406 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST:

a) Minimal: All failed attempts at selecting a user attribute based on the domain of selectable attributes.

b) Basic: All attempts at selecting a user attribute based on the domain of selectable attributes.

c) Detailed: Capture of the values of each user security attribute and domain of selectable attributes mechanism.

## FTA_LSA.1 Limitation on Scope of Selectable Attributes

Hierarchical to: no other components.

**FTA_LSA.1.1** **The TSF shall restrict the scope of the session security attributes [assignment: *session security attributes*], based on [assignment: *attributes*].**

**FTA_LSA.1.2** **Session establishment conditions shall be specifiable only by the authorised administrator.**

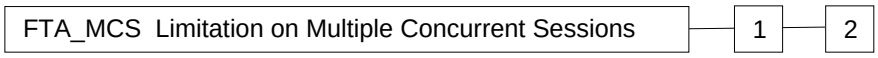
Dependencies : **FIA_ATD.1 User Attribute Definition**
**FTA_TAM.1 Basic TOE Access Management**

## FTA_MCS Limitation on Multiple Concurrent Sessions

Family behaviour

407 This family defines requirements to place limits on the number of concurrent sessions that belong to the same user.

Component levelling




408 FTA_MCS.1 Basic Limitation on Multiple Concurrent Sessions provides limitations that apply to all users of the TSF.

409 FTA_MCS.2 Per User Attribute Limitation on Multiple Concurrent Sessions extends FTA_MCS.1 by requiring the ability for the authorised administrator to specify limitations based on individual user identity and other security attributes.

Audit :

410 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST

a) Minimal: Rejection of a new session based on the limitation of multiple concurrent sessions.

b) Basic: All attempts at establishment of a user session.

c) Detailed: Capture of the number of currently concurrent user sessions and the user security attribute(s).

### FTA_MCS.1 Basic Limitation on Multiple Concurrent Sessions

Hierarchical to: no other components.

**FTA_MCS.1.1 The TSF shall restrict the maximum number of concurrent sessions that belong to the same user.**

**FTA_MCS.1.2 The TSF shall enforce, by default, a limit of a single session per user.**

Dependencies : **FPT_TSA.1 Basic Security Administration**

## FTA_MCS.2 Per User Attribute Limitation on Multiple Concurrent Sessions

Hierarchical to: FTA_MCS.1

**FTA_MCS.2.1** The TSF shall restrict the maximum number of concurrent sessions that belong to the same user **based on [assignment: *security attributes*].**

**FTA_MCS.2.2** The TSF shall enforce, by default, a limit of a single session per user.

**FTA_MCS.2.3** **When more than one user session security attribute is applicable, the TSF shall use [selection: *the minimum number of sessions; the maximum number of sessions*] specified by the set of applicable attributes.**

**FTA_MCS.2.4** **Session establishment conditions shall be specifiable only by the authorised administrator.**

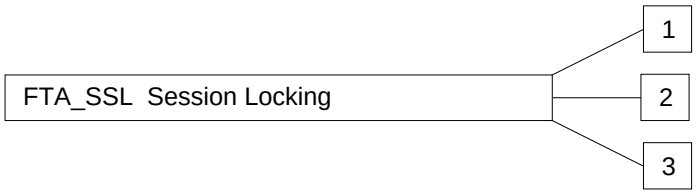
Dependencies : **FIA_UID.1 Basic User Identification**
FPT_TSA.1 Basic Security Administration

# FTA_SSL Session Locking

Family behaviour

411 This family defines requirements for the TSF to provide the capability for TSF-initiated and user-initiated locking and unlocking of interactive sessions.

Component levelling




412 FTA_SSL.1 TSF-Initiated Session Locking includes system initiated locking after a specified period of time.

413 FTA_SSL.2 User-initiated Locking provide capabilities for the user to lock and unlock the user's own interactive sessions.

414 FTA_SSL.3 TSF-initiated Termination provides requirements for the TSF to terminate the session after a period of user inactivity.

Audit :

415 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST:

a) Minimal: Locking of an interactive session by the session locking mechanism.

b) Minimal: Termination of the interactive session by the TSF.

c) Minimal: Successful unlocking of an interactive session.

d) Basic: Any attempts at unlocking an interactive session.

## FTA_SSL.1 TSF-Initiated Session Locking

Hierarchical to: no other components.

**FTA_SSL.1.1 The TSF shall lock an interactive session after a specified interval of user inactivity by:**

**a) clearing or overwriting display devices, making the current contents unreadable;**

**b) disabling any activity of the user's data access/display devices other than unlocking the session.**

**FTA_SSL.1.2** **The default value for the user inactivity interval shall be specifiable only by the authorised administrator.**

**FTA_SSL.1.3** **The TSF shall require user authentication prior to unlocking the session.**

Dependencies : **FTA_TAM.1 Basic TOE Access Management**
**FIA_UAU.1 Basic User Authentication**

### FTA_SSL.2 User-initiated Locking

Hierarchical to: no other components.

**FTA_SSL.2.1** **The TSF shall allow user-initiated locking of the user's own interactive session, by:**

**a) clearing or overwriting display devices, making the current contents unreadable;**

**b) disabling any activity of the user's data access/display devices other than unlocking the session.**

**FTA_SSL.2.2** **The TSF shall require user authentication prior to unlocking the session.**

Dependencies : **FTA_TAM.1 Basic TOE Access Management**
**FIA_UAU.1 Basic User Authentication**

### FTA_SSL.3 TSF-initiated Termination

Hierarchical to: no other components.

**FTA_SSL.3.1** **The TSF shall terminate an interactive session after a specified interval of user inactivity.**

**FTA_SSL.3.2** **The default value for the user inactivity interval shall be specifiable only by the authorised administrator.**

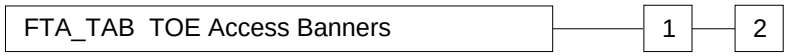
Dependencies : **FTA_TAM.1 Basic TOE Access Management**

# FTA_TAB TOE Access Banners

Family behaviour

416 This family defines requirements to display a configurable advisory warning message to users regarding the appropriate use of the TOE.

Component levelling




417 FTA_TAB.1 Default TOE Access Banners provides a TOE Access Banner that is specified as part of the PP/ST (through the use of the assignment operation). This banner is displayed prior to the establishment dialogue for a session.

418 FTA_TAB.2 Configurable TOE Access Banners provides the ability for the authorised administrator to modify the existing default TOE Access Banner.

## FTA_TAB.1 Default TOE Access Banners

Hierarchical to: no other components.

**FTA_TAB.1.1** **Before establishing a user session, the TSF shall display an advisory warning message regarding unauthorised use of the TOE.**

**FTA_TAB.1.2** **The advisory warning message displayed by the TSF shall be as follows: [assignment: *warning message*] to be displayed.**

Dependencies : No dependencies.

## FTA_TAB.2 Configurable TOE Access Banners

Hierarchical to: FTA_TAB.1

**FTA_TAB.2.1** Before establishing a user session, the TSF shall display an advisory warning message regarding unauthorised use of the TOE.

**FTA_TAB.2.2** The **default** advisory warning message displayed by the TSF shall be as follows: [assignment: *warning message*] to be displayed.

**FTA_TAB.2.3** **The TSF shall restrict the capability to modify the warning message to the authorised administrator.**

Dependencies : **FTA_TAM.1 Basic TOE Access Management**

# FTA_TAH TOE Access History

Family behaviour

419 This family defines requirements for the TSF to display to a user, upon successful session establishment, a history of successful and unsuccessful attempts to access the user's account.

Component levelling


420 There is only one component in this family.

## FTA_TAH.1 TOE Access History

Hierarchical to: no other components.

**FTA_TAH.1.1 Upon successful session establishment, the TSF shall display the [selection: *date, time, method, location*] of the last successful session establishment to the user.**

**FTA_TAH.1.2 Upon successful session establishment, the TSF shall display the [selection: *date, time, method, location*] of the last unsuccessful attempt to session establishment and the number of unsuccessful attempts since the last successful session establishment.**

**FTA_TAH.1.3 The data specified above shall not be removed without user intervention.**

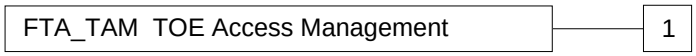
Dependencies : No dependencies.

## FTA_TAM TOE Access Management

### Family behaviour

421 This family defines requirements that enable the authorised administrator to display and modify TOE Access parameters for use in TOE access control.

### Component levelling




422 There is only one component in this family.

### Audit :

423 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST:

a) Minimal: Successful use of the TOE access management function.

b) Basic: All attempts to use TOE access management function; and

c) Basic: Identification of which TOE access parameters have been modified.

d) Detailed: Identification of which TOE access parameters have been modified, with the new values of the parameters.

### FTA_TAM.1 Basic TOE Access Management

Hierarchical to: no other components.

**FTA_TAM.1.1 The TSF shall restrict the capability to display and modify TOE Access parameters to the authorised administrator.**

**FTA_TAM.1.2 The TSF shall allow the authorised administrator the flexibility to display all [selection: *TOE Access parameters for a user, users associated with a TOE Access parameter*].**

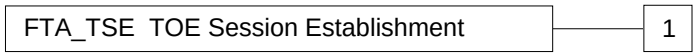
Dependencies : **FPT_TSA.1 Basic Security Administration**

# FTA_TSE TOE Session Establishment

Family behaviour

424 This family defines requirements to allow or deny an user to establish a session with the TOE based on environmental conditions.

Component levelling




425 There is only one component in this family.

Audit :

426 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST:

a) Minimal: Successful use of the session establishment mechanism.

b) Basic: All attempts at establishment of a user session.

c) Detailed: Capture of the value of the selected access parameters (e.g., location of access, time of access).

## FTA_TSE.1 TOE Session Establishment

Hierarchical to: no other components.

**FTA_TSE.1.1 The TSF shall be able to deny session establishment based on [assignment: *attributes*].**

**FTA_TSE.1.2 Session establishment conditions shall be specifiable only by the authorised administrator.**

Dependencies : **FIA_ATD.1 User Attribute Definition**
**FTA_TAM.1 Basic TOE Access Management**

# Class FTP

# Trusted Path/Channels

426 Families in this class provide requirements for a trusted communication path between users and a TSF, and for a trusted communication channel between TSFs that have the following general characteristics:

- The communications path is constructed using internal and external communications channels (as appropriate for the component) that isolate an identified subset of TSF data and commands from the remainder of the TSF and user data.
- Use of the communications path may be initiated by the user and/or the TSF (as appropriate for the component)
- The communications path is capable of providing assurance that the user is communicating with the correct TSF, and that the TSF is communicating with the correct user (as appropriate for the component)

426 In this paradigm, a ***trusted channel*** is a communication channel that may be initiated by either side of the channel, and provides non-repudiation characteristics with respect to the identity of the sides of the channel.

427 A ***trusted path*** provides a means for users to perform functions through an assured direct interaction with the TSF. Trusted path is usually desired for user actions such as initial identification and/or authentication, but may also be desired at other times during a user's session. Trusted path exchanges may be initiated by a user or the TSF. User responses via the trusted path are guaranteed to be protected from undetected modification by untrusted applications.

428 Absence of a trusted path may allow breaches of accountability or access control in environments where untrusted applications are used. These applications can intercept user-private information, such as passwords, and use it to impersonate those users. As a consequence, responsibility for system actions cannot be reliably assigned to an accountable entity. Also, these applications could output erroneous information on an unsuspecting user's display, resulting in subsequent user actions that may be erroneous and may lead to a security breach.

429 Figure 2.17 shows the decomposition of this class into its constituent components.

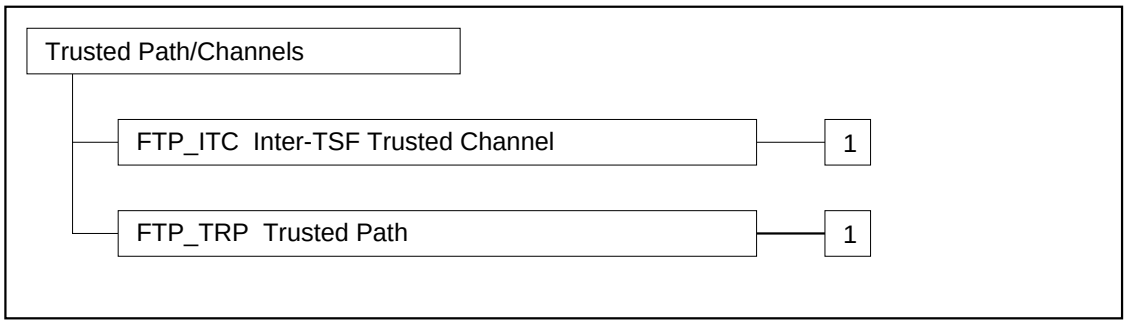


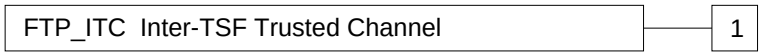

**Figure 2.17 - Trusted Path / Channels Class decomposition**

# FTP_ITC Inter-TSF Trusted Channel

Family behaviour

430 This family defines requirements for the creation of a trusted channel between the TSF and other TSFs for the performance of security critical operations. This family should be included whenever there are requirements for the secure communication of user or TSF data between two TSFs.

Component levelling




431 FTP_ITC.1 Inter-TSF Trusted Channel requires that the TSF provide a trusted communication channel between itself and another TSF.

Audit :

432 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: Successful use of the trusted channel functions.

b) Minimal: Identification of the initiator and target of the trusted channel.

c) Basic: All attempted uses of the trusted channel functions.

## FTP_ITC.1 Inter-TSF Trusted Channel

Hierarchical to: no other components.

**FTP_ITC.1.1 The TSF shall provide a communication channel between itself and a remote TSF that is logically distinct from other communication channels and provides assured identification of its endpoints.**

**FTP_ITC.1.2 The [selection: *TSF shall have, remote TSF shall be allowed*] the ability to initiate communication via the trusted channel.**

**FTP_ITC.1.3 The TSF shall initiate the trusted channel for [assignment: *list of functions for which a trusted channel is required*].**

Dependencies : No dependencies.

# FTP_TRP Trusted Path

Family behaviour

433 This component defines the requirements to establish and maintain trusted communication to or from human users and the TSF. A trusted path may be required for any security-relevant interaction. Trusted path exchanges may be initiated by a human user during an interaction with the TSF, or the TSF may establish communication with the human user via a trusted path.

Component levelling


434 FTP_TRP.1 Trusted Path requires that a trusted path between the TSF and a human user be provided for a PP/ST author defined set of events. The user and/or the TSF may have the ability to initiate the trusted path.

Audit :

435 The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP / ST:

a) Minimal: Successful use of the trusted path functions.

b) Basic: All attempted uses of the trusted path functions.

c) Basic: Identification of the initiator and target of the trusted path.

## FTP_TRP.1 Trusted Path

Hierarchical to: no other components.

**FTP_TRP.1.1 The TSF shall provide a communication path between itself and [selection: *remote, local*] human users that is logically distinct from other communication paths and provides assured identification of its endpoints.**

**FTP_TRP.1.2 [selection: *The TSF, local users, remote users*] shall have the ability to initiate communication via the trusted path.**

**FTP_TRP.1.3 The TSF shall require the use of the trusted path for [selection: *initial user authentication*, [assignment: *Other services for which trusted path is required*]].**

Dependencies : No dependencies.

## Chapter 3

# Predefined functional packages

*Editor Note:* ***In the next issue of the CC this annex will contain agreed functional packages based upon functional requirements from the source criteria documents (ITSEC, CTCPEC, FC / TCSEC).***